Secure Cloud-Connected Robot Control using Private Blockchain

Muhammad Amzie Muhammad Fauzi[1], Mohamad Hanif Md Saad[2], Sallehuddin Mohamed Haris[3], Marizuana Mat Daud[4]
Faculty of Engineering & Built Environment, Universiti Kebangsaan Malaysia, Selangor, Malaysia[1, 2, 3]
Institute of IR4.0, Universiti Kebangsaan Malaysia, Selangor, Malaysia[4]

***Abstract*—With the increasing demand for remote operations and the challenges posed during the COVID-19 pandemic, industries across various sectors, including logistics, manufacturing, and education, have adopted virtual solutions. Cloud-based robot control has emerged as a viable approach for enabling safe remote operation of robots. However, along with the benefits, there are also risks associated with cloud-based robots. In this study, a secure cloud-based robot control system using a blockchain system was developed. The robot utilizes supervisory control to navigate via the internet. The communication system of the robot relies on the ThingsSentral cloud-based IoT platform; enabling communication between the user via a GUI developed using Python Tkinter and the local robot over the internet. To facilitate internet communication, the robot in this study incorporates an ESP32 microcontroller, which provides a low-cost and low-power system capable of connecting to Wi-Fi. However, cloud-based control systems are susceptible to cyberattacks, prompting the use of blockchain cybersecurity in this study to mitigate the risks. The data sent by the supervisor is stored within a private blockchain developed using Python, simultaneously being transmitted to the cloud platform. The developed security system addresses the risks associated with cloud-based robot control systems, such as data tampering and unauthorized misuse, by leveraging the Proof of Work (PoW) and hashing mechanisms.**

Keywords*—*Internet of Things (IoT); robot control; cloud computing; cybersecurity; blockchain

I. Introduction

Industries from a variety of sectors have adopted virtual ways of conducting business in these challenging times due to the enormous problems posed by social distance measures. As a result, there is a huge increase in demand for robots that can be operated safely and remotely for practical research in academic settings. The ability to securely and easily control robots from a distance has become possible thanks to cloud-based interconnection, assuring the continuity of hands-on learning experiences [1].

Cloud-based robot control refers to the practice of remotely controlling robots through cloud computing infrastructure. In order to enable effective and scalable control of robotic systems, it includes utilizing the cloud's capabilities and resources [2]. The robot offloads some duties or calculations to the cloud, which can offer improved computational capabilities, storage, and network connectivity, as opposed to merely depending on onboard processing and control systems [3]. Nowadays, autonomous vehicles or cloud-based robots are mostly used in the service sector. There are many different kinds of service robots, including robots for cleaning [4], [5] and housekeeping [6], surveillance [7], entertainment [8], rehabilitation [9], and so forth [10]. In Malaysia, service robots have been widely used in restaurants to deliver food to customers' tables [11].

These days, residential robots are also substantially included in service robots. In smart homes, restaurants, and hospitals, it is crucial [12], [13]. Roombas [14], vacuum cleaners [5], [15], lawnmowers [16], [17], waiter robots [18], security robots, and sentry robots [19], [20] are a few examples of residential robots. Domestic service robots are becoming more and more common because of how convenient they are and how much time they can save homeowners for other things. To improve their usefulness and performance, they frequently make use of cutting-edge sensors, artificial intelligence, and cloud connectivity. Service robots are becoming more and more popular, which will lead to concerns regarding their security.

Because robot systems are networked and rely on software control, guaranteeing cybersecurity is essential in this field. However, because they are primarily focused on other issues, many initiatives fail to take into account how susceptible robot control systems are to cyberattacks. Frequently, projects create their own cloud connections without paying enough attention to security, potentially creating holes in coding frameworks and other parts. Robots are integrated with the Internet as cyber-physical systems, making them vulnerable to different cybersecurity threats like malware, phishing schemes, and illegal access [21]. Businesses and individuals must prioritize cybersecurity precautions, such as using safe passwords, keeping software updated, and utilizing antivirus and anti-malware software, in order to reduce these risks.

Robots that are autonomous and connected to the cloud have the risk of being misused by authorized users in addition to being exploited by hackers. Robots connected to the cloud execute user-issued orders. A security measure needs to be implemented on the robot to ensure that the commands being delivered are coming from the intended user. Blockchain technology can assist in tracking the actions taken in response to commands provided through the system's digital ledger. The technology of blockchain is now gaining a lot of interest and may help with IoT security challenges. Due to its decentralized architecture, ability to provide data immutability, and non-repudiation services, blockchain technology appears to be a promising approach for securing IoT and protecting user/data privacy.

The advantages of blockchain technology are dependability, security, and efficiency. By limiting access to authorized users, it guarantees the timeliness and accuracy of information for users within a members-only network. A blockchain transaction cannot be changed after it has been recorded, not even by administrators. Time-consuming reconciliations are no longer necessary thanks to the distributed ledger, and smart contracts automate transaction execution [22]. Block-based transactional data is stored in decentralized databases called blockchains. Before being uploaded to the blockchain, each transaction is first checked for accuracy and encrypted by the parties involved or trusted nodes. Depending on the access and encryption methods used, blockchains can be either public or private [23]. There are lots of public blockchain used currently, mainly used in cryptocurrency transactions such as Bitcoin [24] and Ethereum [25] while Hyperledger Fabric [26] is an example of private blockchain.

In this paper, we present a secure cloud-based robot control using blockchain. The robot’s control system allows it to navigate toward a designated and specified location while having a security system to prevent cyberattacks such as data breaches and misuse by authorized users. The paper is organized as follows: The next section explains related works on the types of robot control system and cybersecurity of cloud-based robot. The following section presents the steps taken to complete this study, the IoT platform used in this paper (The ThingsSentral), and the development of a supervisory control system. Then, the following section covers the implementation and testing of the cloud-based robot control system, and a performance test analysis of the supervisory control system using blockchain technology. The last section gives the conclusion and future work of the research.

II. Related Works

A. Robot Control Systems

There are numerous ways to control service robots. The control system that is chosen depends on the particular needs of the service robot application for which it is intended. For instance, on-board control systems, LAN control systems, and cloud-based control systems can all be utilized to operate service robots [27]–[29]. The types of control systems and a few instances of their use are presented in Fig. 1.

1) On-board control system: A control system called an on-board control system is one that is built into the robot's hardware. The robot has the ability to operate without external systems. This kind of control system is widely used by autonomous service robots, which are robots that can travel and do tasks on their own thanks to sensors and algorithms. Restaurant robots, cleaning robots, logistic robots, and social robots are a few examples of robots with on-board control systems.

In comparison to cloud-based service robots, on-board control systems can offer real-time reactivity, a high degree of autonomy, less latency or buffering, and better security. To execute the command without error, applications of service robots employing on-board control systems must be controlled without buffer [6], [30]. ROS has some restrictions even though it can be used as an on-board control system. Fully autonomous robots may have several advantages for the user [31], although initial setup may be laborious. The kinematics computation may be aided by the use of coding, such as MATLAB [32], to carry out a command, such as grabbing an object with a robotic arm.

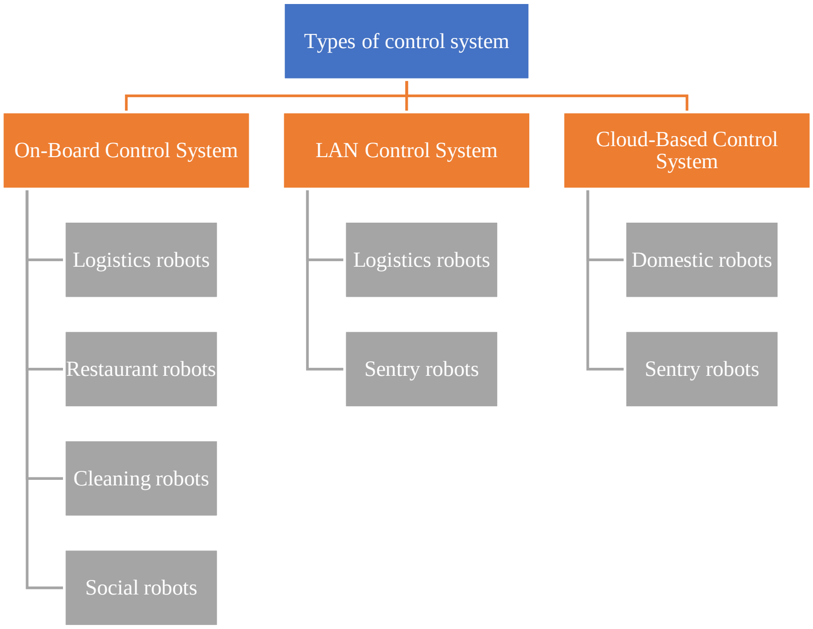


Fig. 1. Types of robot control system.

2) LAN control system: In a LAN (Local Area Network) control system, the robot is connected to a control station or a remote device over a local area network. This approach is frequently used by industrial robots and mapping sentry robots [28]. The robot receives commands from the control station via the LAN and reacts with sensor data and status updates.

LAN management mechanism service robots are machines with remote controls connected to a local network. Due to their connection to a single network, LAN control system service robots may have various benefits over cloud-based and on-board control systems, including reduced latency, high levels of autonomy, and improved security. However, the drawbacks of using LAN to control the service robot are limited external connectivity and unreliable network conditions [33]. This can be avoided by sending data to the server across multi-layered networks or by adding additional relay nodes [34], [35]. This strategy could be laborious, though, as the system's complexity and price will skyrocket.

3) Cloud-based control system: In order to remotely control and manage the behaviour of a service robot, the control system uses cloud computing technology. Domestic robots and sentry robots are a couple of the uses for cloud-based service robots [29]. By fusing the robot's control system with cloud-based servers, this approach enables the robot to quickly access and process enormous amounts of data.

The use of cloud computing in standalone and remotely networked mobile robot systems is the current focus of robotics research. Real-time robot path planning with cloud-based, computationally expensive evolutionary algorithms [36]. Recent studies show that cloud-based control system using IoT platform is increasingly popular. 46 IoT platforms have been described as open, whereas 25 systems are deemed to be open by some research but not by the platforms themselves [37]. In this study, the cloud platform used is The ThingsSentral cloud platform [38]. ThingsSentral is a cloud platform developed by the CAISER research group at the Faculty of Engineering and Built Environment (FKAB), Universiti Kebangsaan Malaysia (UKM).

B. Robot Cybersecurity

Service robot cyber security refers to the safeguarding of service robots against cyberthreats such as malware, hacking, and other cyberattacks. As service robots become more tightly connected to the internet and other devices, the potential of hacks increases [39]. Cyberattacks can take many different forms. DDoS is the most typical [40]. The cloud robot will cease operations if a DDoS attack is launched against its server, which could cause a significant loss [41]. When building service robots [42], secure coding techniques ought to be applied to minimize the risk of vulnerabilities.

1) Authentication: Strong authentication mechanisms should be used to increase the security of service robots [43]. It is essential to make sure that these robots are only accessible to those who are allowed. Passwords, biometric authentication [44], face recognition [45], and other authentication techniques can all be used. Users must supply a legitimate password or passphrase for password-based authentication in order to prove their identity. Biometric identification verifies users by using distinctive biological traits like fingerprints or iris patterns. To authenticate people, face recognition technologies compare and evaluate facial traits. Service robots can build strong security standards, prohibiting unwanted access, and guaranteeing that only authorized users can control and interact with the robots.

2) Encryption: Data encryption is another method to improve the cyber-security of service robots in addition to authentication. Communication between service robots and their control systems should be encrypted to prevent unauthorized data interception [46]. Secure communication technologies that can be used for this include SSL/TLS and VPN. The usage of service robots involves the use of numerous sensors to collect a lot of data, some of which could be sensitive. Through the cloud service, the data is sent to a communication protocol [47]. In order to lower the danger of cyberattacks, data transfer encryption must be robust. Users' sensitive data can be well-protected by using encrypted data.

3) Blockchain: There is only a few research that has been done on cloud-based service robots with blockchain technology as the complexity of the system is high. However, robotics and cloud computing relate highly to one another and are becoming increasingly important as both technologies evolve in the IR4.0 applications. Robotic system management and deployment platforms can be made available by cloud computing. Robotic systems can be centralized handled and watched, allowing for real-time changes and upgrades, by leveraging cloud-based software platforms. Some of these applications are in the domains of smart city and smart industry [48]. Therefore, we can also say that blockchain and cloud computing relate to one another with blockchain and robotics.

Blockchain is used in cloud computing to build decentralized storage solutions that are safe [49]. Blockchain-based storage, in contrast to conventional centralized cloud storage, eliminates single points of failure and potential hacker targets [50]. Blockchain improves security and attack resilience by dispersing data across a decentralized network. Blockchain makes it possible to create secure and open identity management systems in the context of robotics and cloud computing [51]. Users can securely validate their identities when using cloud services through decentralized and tamper-proof records [52], [53]. Blockchain's incorporation into cloud computing provides greater data security, less chances of data tampering, and improved identity verification processes. Fig. 2 shows the basic operation of Blockchain.

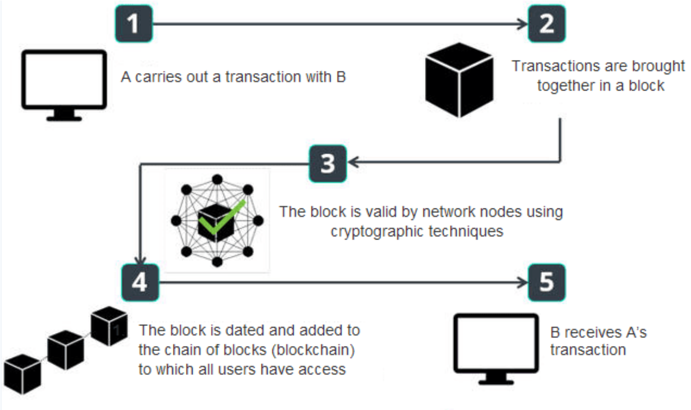

1
A carries out a transaction with B
2
Transactions are brought together in a block
3
The block is valid by network nodes using cryptographic techniques
4
The block is dated and added to the chain of blocks (blockchain) to which all users have access
5
B receives A's transaction

Fig. 2. Basic process of blockchain adopted from (URL:- https://www.centralcharts.com/en/gm/1-learn/1-cryptocurrency/42-trading/699-definition-of-blockchain).

III. Methodology

Fig. 3 depicts the main flowchart of this study. The study begins with the construction of robot using microcontroller, actuators, sensors and power supplies. This study continues with the development of the robot communication system with the cloud platform ThingsSentral. After achieving satisfactory result of the testing of communication system developed, this study continues to the development of GUI control system. The results achieved in this study depend on the GUI developed and also the cybersecurity implemented in this study.

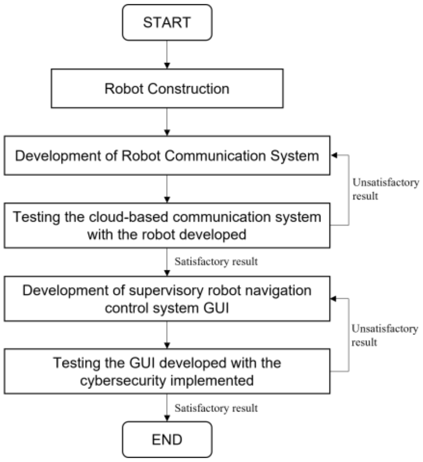

START
Robot Construction
Development of Robot Communication System
Unsatisfactory result
Testing the cloud-based communication system with the robot developed
Satisfactory result
Development of supervisory robot navigation control system GUI
Unsatisfactory result
Testing the GUI developed with the cybersecurity implemented
Satisfactory result
END

Fig. 3. Flowchart of the study.

A. Development of Robot Communication System

This study uses cloud-based control system as its main communication system. The robot developed, which is located at FKAB, UKM must have the ability to be controlled from a base station located elsewhere. This study uses ThingsSentral cloud platform to send the command from the base station to the robot. The data sent from the supervisor is stored inside the blockchain at the same time it is sent towards the cloud platform. This ensures the command sent by the supervisor as blockchain transactions. Supervisors can also read data from cloud platform such as latest data or distance. The robot reads the data from cloud and sends sensor data to ThingsSentral to be read by the supervisor. Fig. 4 shows the framework of this study.

The software used to develop the communication system between the robot and ThingsSentral cloud platform is Arduino IDE and Python. Both the Arduino IDE and Python software communicate to ThingsSentral web application using Web API. Fig. 5 shows the ThingsSentral web application that was used in this study.

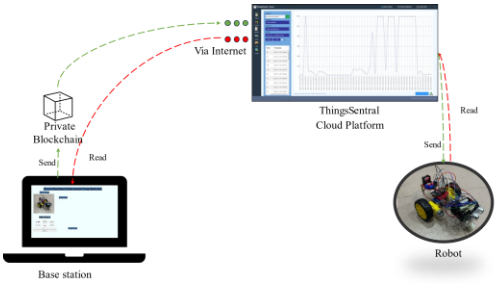


Fig. 4. Framework of the study.

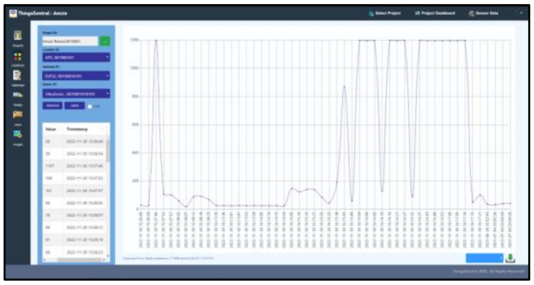


Fig. 5. ThingsSentral web application.

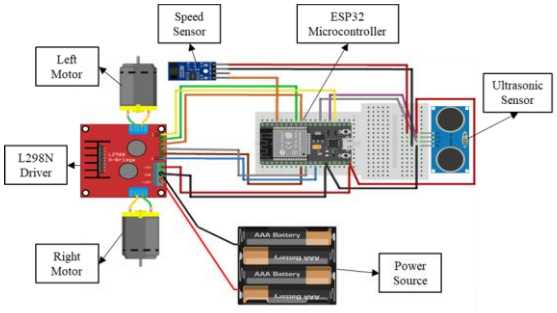


Fig. 6. Schematic diagram of robot.

The robot developed in this study relies on batteries or direct current as its primary power source. The batteries supply power to activate the motors, which are connected to the tires, microcontroller, and sensors. The microcontroller chosen for this research is the ESP32, which utilizes program memory and data memory to store commands and data. Fig. 6 illustrates the schematic diagram of the robot utilized in this study, featuring the ESP32 microcontroller, actuators, and sensors such as ultrasonic and speed sensors. For communication between the robot and the user as a supervisory control system, the ThingsSentral platform is employed. This cloud-based control system allows the robot to be moved remotely and wirelessly. To facilitate control over the robot, a web GUI application has been developed using Python software. This application enables the user to control the robot's actuators and obtain real-time sensor data from the robot.

The data exchange between the ESP32 microcontroller and the cloud platform ThingsSentral is facilitated through an intermediary platform—a GUI developed using Python. This communication system allows for the sending and receiving of data via ThingsSentral. The adoption of cloud-based control systems for service robots is dependent on the specific requirements and constraints of each application. ThingsSentral offers a security system to users, ensuring that the communication mechanism between the user and the robot can be safely controlled without the risk of unauthorized manipulation by attackers.

Moreover, ThingsSentral enables users to control the developed robots over a different network. This means that the robot can be operated from various locations, regardless of the user's distance, as long as there is a stable internet connection [54]. In this study, a cloud platform is essential for the communication system between the robot and the controller because it provides a dedicated static IP address for sending and receiving user data. This differs from a local area network (LAN) control system, as the robot requires its own IP address to ensure proper functioning of the communication system when a cloud platform is not available to assign the IP address. However, it is important to note that the availability of static IP addresses for the robot is limited, which presents some limitations to this solution.

Cloud-based control systems offer numerous advantages, but they also come with drawbacks, including latency and dependence on external networks. The focus of this study is on addressing the security and privacy concerns associated with such systems. While ThingsSentral provides a security system for users, it is still susceptible to misuse by authorized individuals. Hence, the objective of this study is to develop a cybersecurity system to prevent the misuse of the control system by authorized users. In this study, the chosen cybersecurity solution is Blockchain. The reason for selecting Blockchain is its inherent immutability and permanence. Transactions recorded in a Blockchain system cannot be altered or deleted, as they are permanently stored within the system. This ensures the accuracy of every transaction, and users are unable to assign blame to other parties for commands issued by themselves. By leveraging Blockchain, the study aims to enhance the security and integrity of the control system.

B. *Secure Cloud-based Robot Control Implementation and Testing*

Fig. 7 illustrates the robot prototype utilized in this study, which was developed based on the schematic diagram depicted in Fig. 6. The communication of data is facilitated through the ESP32 microcontroller, serving as the processor for communication between the robot and the ThingsSentral cloud platform. The robot itself is equipped with two tires, each connected to a DC motor, an L298N driver responsible for motor direction control, sensors including an encoder and an ultrasonic sensor, and AA batteries serving as the power source. The microcontroller handles the reading and sending of data to the ThingsSentral web application. To enable this transmission, an intermediary (API) is required to establish a connection between the gateway and the internet.

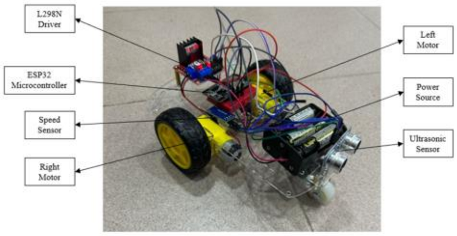


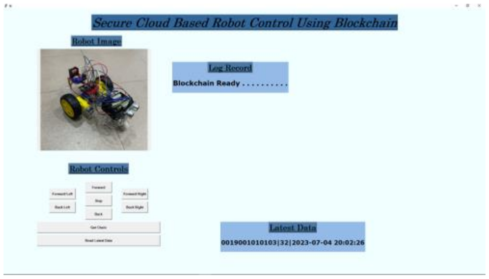
Fig. 7. Robot prototype used in this study.

In this study, HTTP protocols are utilized for both data transfer and data ingestion from ThingsSentral. The sensor nodes receive the transmitted data, which includes the distance measured by the ultrasonic sensor and the motor speed obtained from the encoder. The Arduino IDE software is employed to facilitate data transmission from the sensor nodes to the ThingsSentral cloud platform, as the data is stored within the ESP32 microcontroller. Through programming, the data collected by the sensor nodes can be transmitted to ThingsSentral using URLs and internet networks.

C. *Development of Supervisory Control System GUI*

Fig. 8 showcases the developed supervisory control system created using Python Tkinter in this study. The graphical user interface (GUI) includes buttons that have been programmed to transmit data values to ThingsSentral, which serve as instructions to control the robot's motors. For example, when the "Stop" button is pressed, it sends a command value of 0. Each motor will be assigned a value of 0 and 1 depending on the state of the motor after the command value is sent. This command value is then utilized by the ESP32 microcontroller to instruct the motors to halt the robot's movement. Each button on the GUI corresponds to a different command value, enabling control over the robot's navigation.

Furthermore, in addition to the buttons used to send command values for robot navigation, the developed GUI also displays the most recent sensor readings obtained from the node sensor via the ThingsSentral cloud platform. The node sensor, located at the bottom right of Fig. 8, represents the ultrasonic sensor. To obtain and display the latest sensor reading, the HTTP Get protocol is employed to retrieve data from ThingsSentral. The ultrasonic reading provides information to the user regarding the distance between the robot and any obstacles in front of it. An example of the value sent by the GUI to the ThingsSentral cloud platform is presented in Table I.



```
Blockchain Ready ..........
```

```
0019001010103|32|2023-07-04 20:02:26
```

Fig. 8. Supervisory control system.

IV. Results and Discussion

A. Results

The timestamp provided in Table I depicts when the command value was received by ThingsSentral cloud platform. However, the time may differ from when the command value was sent by the supervisory GUI as buffer is present when controlling the robot through cloud-based control system. The difference in time depends on factors such as internet speed and the load sent by the GUI. Additionally, Fig. 9 presents a graphical representation of the command values for each motor plotted against the timestamp.

After successfully testing the supervisory control system's performance, the security of the robot control system underwent evaluation. Employing Python's hashlib module, this study crafted a blockchain system. The command payload from the supervisory control system is incorporated into a block generated by the module. This block includes an index indicating its position in the chain, the block's hash, the timestamp of the sent payload, the payload data specifying the robot's movement command, proof of work, and the previous block's hash. Utilizing the SHA-256 algorithm in the hashlib module, the block's validity, determined by its hash, proof of work, and the previous block's hash, is verified.

TABLE I. Command Value of each Motor

#	Desired Navigation	Command Value	Left Motor	Right Motor	Timestamp
1	Stop	0	0	0	28/6/2023 17:11:24
2	Forward	1	1	1	28/6/2023 17:11:26
3	Forward Left	5	0	1	28/6/2023 17:11:28
4	Forward	1	1	1	28/6/2023 17:11:30
5	Stop	0	0	0	28/6/2023 17:11:32
6	Backward Right	4	-1	0	28/6/2023 17:11:34
7	Backward	2	-1	-1	28/6/2023 17:11:37
8	Stop	0	0	0	28/6/2023 17:11:39
9	Forward Right	3	1	0	28/6/2023 17:11:47
10	Stop	0	0	0	28/6/2023 17:11:50

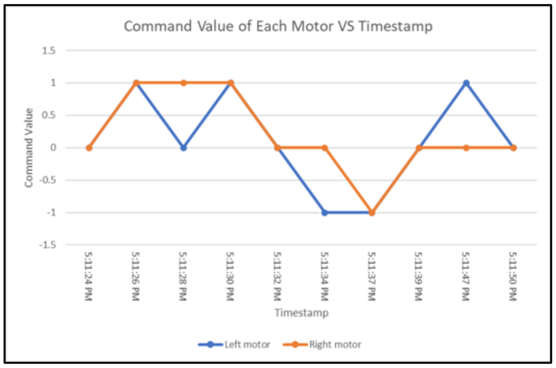
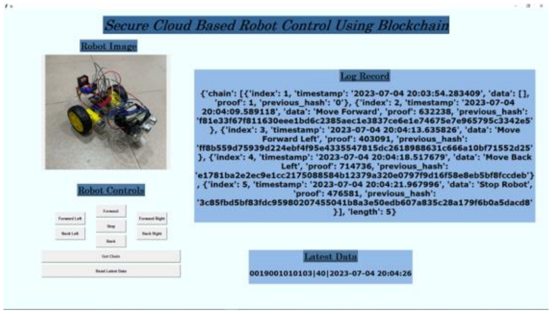


Fig. 9. Graph of command value of each motor Vs timestamp.

The GUI allows for the display of the chain by pressing the "Get Chain" button, as depicted in Fig. 10. The chain is presented on the right side of the GUI, showcasing the total length of the chain and listing the blocks. Each block includes information such as the index number, timestamp, payload (determined by the button pressed to navigate the robot), PoW, and the previous hash. The latest data on the bottom right of Fig. 10 indicates the latest ultrasonic data read from ThingsSentral cloud platform. This data depicts the distance between the robot and an obstacle in front of it.



```
{'chain': [{'index': 1, 'timestamp': '2023-07-04 20:03:54.283409', 'data': [],
'proof': 1, 'previous_hash': '0'}, {'index': 2, 'timestamp': '2023-07-04
20:04:09.589118', 'data': 'Move Forward', 'proof': 632238, 'previous_hash':
'f81e33f67f811630eee1bd6c2385aec1e3837ce6e1e74675e7e965795c3342e5'
}, {'index': 3, 'timestamp': '2023-07-04 20:04:13.635826', 'data': 'Move
Forward Left', 'proof': 403091, 'previous_hash':
'ff8b559d75939d224ebf4f95e4335547815dc2618988631c666a10bf71552d25'
}, {'index': 4, 'timestamp': '2023-07-04 20:04:18.517679', 'data': 'Move Back
Left', 'proof': 714736, 'previous_hash':
'e1781ba2e2ec9e1cc2175088584b12379a320e0797f9d16f58e8eb5bf8fccdeb'}
, {'index': 5, 'timestamp': '2023-07-04 20:04:21.967996', 'data': 'Stop Robot',
'proof': 476581, 'previous_hash':
'3c85fbd5bf83fdc95980207455041b8a3e50edb607a835c28a179f6b0a5dacd8'
}], 'length': 5}
```

```
0019001010103|40|2023-07-04 20:04:26
```

Fig. 10. Blockchain system in this study.

B. Discussion

The supervisory control system, developed using Python software, is connected to the ESP32 microcontroller through programmed code using Arduino. This connection is facilitated by making ThingsSentral serve as an intermediary. The control system sends data to the cloud, which is then read by the robot via the microcontroller to execute the issued commands. The value of the data sent by the GUI determines the direction of the robot's navigation, instructing the motor driver to activate the robot's motors. The communication of data between the robot and the ThingsSentral cloud platform is achieved using the HTTP GET protocol. Upon receiving the command value, the ESP32 utilizes it as an instruction to engage the motor driver, which controls the robot's movement by directing the motor's speed and rotational direction.

Every command value corresponds to a specific action for the motors, determining the robot's movement or halt according to the intended navigation. For example, if the supervisor instructs the robot to move forward, each motor will be assigned a value between 0 and 1, depending on the desired direction. A positive value indicates forward movement, while a negative value signifies backward motion. Table I provides an overview of the command values assigned to each motor based on the desired navigation.

The security system developed in this study is based on Blockchain technology. Blockchain offers data tampering and breach prevention mechanisms through its Proof of Work (PoW) and hashing mechanisms. PoW serves as a consensus mechanism in Blockchain networks, creating a robust barrier against data breaches. To modify a block within the blockchain, an attacker would need to alter the block's contents and recalculate the hash, which is computationally expensive and time-consuming due to PoW. Additionally, rewriting the entire blockchain would require an attacker to possess the majority of the network's computational power, making it highly improbable.

The blockchain system developed in this study utilizes the hashlib module in Python software. This module provides a universal interface for various secure cryptographic hash and message digest algorithms. Each hash algorithm has its designated constructor method, creating a hash object with a straightforward interface. The hashlib module supports several hash algorithms, including Secure Hash Algorithm 1 (SHA-1), SHA-224, SHA-256, SHA-384, SHA-512, and MD5. These functions generate fixed-size hash values, known as the hash value or digest, based on the input data they receive.

V. Conclusion

This research presents the development of a secure cloud-based robot control system incorporating blockchain technology. Utilizing an ESP32 microcontroller and Wi-Fi connectivity to ThingsSentral, the robot aims to navigate specific desired locations through a cloud-based control system. The implementation involves Python Tkinter for a user-friendly GUI with dedicated buttons, serving as controllers for robot navigation. The GUI facilitates data transmission to the microcontroller via ThingsSentral, serving as an intermediary cloud platform. However, inherent risks of data tampering and misuse in cloud-based control systems are acknowledged. To counter these concerns, the study introduces a blockchain security system, leveraging hashing, Proof of Work (PoW), and transaction records. Through these mechanisms, the study aims to mitigate the identified risks and enhance overall cybersecurity in the cloud-based robot control system. While this study provides valuable insights, limitations include exclusive use of private blockchain technology visible only to certain organization members, raising concerns about potential transaction payload changes that may impact user confidence in system security. Future research may explore leveraging public blockchains like Ethereum for enhanced transaction verification, offering transparency and validation through universal observation by a diverse user base.

Acknowledgment

The authors would like to thank Universiti Kebangsaan Malaysia for supporting this research. We would also like to thank the Faculty of Engineering and Built Environment and the Institute of IR4.0, UKM for providing the research facilities.

References

[1] Y. Wei, S. Hong, M. Ma, J. Xie, Z. Lu, and X. Zheng, "Raspberry Pi 4B-based cloud-based robot design and demonstration platform construction," 2023 IEEE 3rd Int. Conf. Power, Electron. Comput. Appl. ICPECA 2023, pp. 1736–1739, 2023, doi: 10.1109/ICPECA56706.2023.10076246.

[2] Y. Wang, N. Wang, Z. Chen, and W. Chen, "A fully cloud-based modular home service robot," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 10464 LNAI, no. 100, pp. 320–334, 2017, doi: 10.1007/978-3-319-65298-6_30.

[3] G. Tian, H. Chen, and F. Lu, "Cloud computing platform based on intelligent space for service robot," 2015 IEEE Int. Conf. Inf. Autom. ICIA 2015 - conjunction with 2015 IEEE Int. Conf. Autom. Logist., no. August, pp. 1562–1566, 2015, doi: 10.1109/ICInfA.2015.7279535.

[4] R. Wang, H. Guan, and J. Lan, "Home services - Track transport robot control system design," Appl. Mech. Mater., vol. 341–342, pp. 646–649, 2013, doi: 10.4028/www.scientific.net/AMM.341-342.646.

[5] Z. Wei, "Design of Control System for Dust-Collecting Robot Based on DSP," Proc. - 2017 Int. Conf. Comput. Network, Electron. Autom. ICCNEA 2017, vol. 2017-Janua, pp. 437–440, 2017, doi: 10.1109/ICCNEA.2017.85.

[6] M. Kim, T. Kang, D. Song, and S. J. Yi, "Development of a small-sized intelligent home service robot," 2021 18th Int. Conf. Ubiquitous Robot. UR 2021, pp. 565–570, 2021, doi: 10.1109/UR52253.2021.9494667.

[7] M. H. Md Saad, "Room Searching Performance Evaluation for the JagaBotTM Indoor Surveillance Robot," KnE Eng., vol. 1, no. 2015, pp. 1–6, 2016, doi: 10.18502/keg.v1i1.486.

[8] H. S. Ahn et al., "Entertainment services of a healthcare robot system for older people in private and public spaces," ICARA 2015 - Proc. 2015 6th Int. Conf. Autom. Robot. Appl., pp. 217–222, 2015, doi: 10.1109/ICARA.2015.7081150.

[9] B. Li, G. Li, Y. Sun, G. Jiang, J. Kong, and D. Jiang, "A review of rehabilitation robot," Proc. - 2017 32nd Youth Acad. Annu. Conf. Chinese Assoc. Autom. YAC 2017, pp. 907–911, 2017, doi: 10.1109/YAC.2017.7967538.

[10] G. Fabregat, J. A. Belloch, J. M. Badia, and M. Cobos, "Design and Implementation of Acoustic Source Localization on a Low-Cost IoT Edge Platform," IEEE Trans. Circuits Syst. II Express Briefs, vol. 67, no. 12, pp. 3547–3551, 2020, doi: 10.1109/tcsii.2020.2986296.

[11] "The rise of robot waiters in Malaysia, driven by labour shortage in F&B sector | The Star." https://www.thestar.com.my/lifestyle/living/2022/08/24/the-rise-of-robot-waiters-in-malaysia (accessed Mar. 02, 2023).

[12] Y. Shi and H. Fan, "Research on structure design and kinematics equation of restaurant service robot manipulator," Adv. Mater. Res., vol. 490–495, pp. 2700–2703, 2012, doi: 10.4028/www.scientific.net/AMR.490-495.2700.

[13] Z. Zhaohui, X. Mei, X. Bian, H. Cai, and J. Ti, "Development of an intelligent interaction service robot using ROS," Proc. 2016 IEEE Adv. Inf. Manag. Commun. Electron. Autom. Control Conf. IMCEC 2016, pp. 1738–1742, 2017, doi: 10.1109/IMCEC.2016.7867516.

[14] E. Ruiz, R. Acuña, N. Certad, A. Terrones, and M. E. Cabrera, "Development of a control platform for the mobile robot Roomba using ROS and a Kinect sensor," Proc. - 2013 IEEE Lat. Am. Robot. Symp. LARS 2013, pp. 55–60, 2013, doi: 10.1109/LARS.2013.57.

[15] K. M. Hasan, Abdullah-Al-Nahid, and K. J. Reza, "Path planning algorithm development for autonomous vacuum cleaner robots," 2014 Int. Conf. Informatics, Electron. Vision, ICIEV 2014, pp. 1–6, 2014, doi: 10.1109/ICIEV.2014.6850799.

[16] A. V. Proskokov, M. V. Momot, D. N. Nesteruk, E. S. Terentyev, and A. D. Veretennikov, "Software and Hardware Control Robotic Lawnmowers," J. Phys. Conf. Ser., vol. 1059, no. 1, 2018, doi: 10.1088/1742-6596/1059/1/012018.

[17] M. Ryalat, M. Alsherqatli, and H. Elmoaqet, "IoT-aided Smart Lawnmower," ACM Int. Conf. Proceeding Ser., 2019, doi: 10.1145/3386164.3387298.

[18] T. M. N. U. Akhund, M. A. B. Siddik, M. R. Hossain, M. M. Rahman, N. T. Newaz, and M. Saifuzzaman, "IoT Waiter Bot: A Low Cost IoT based Multi Functioned Robot for Restaurants," ICRITO 2020 - IEEE 8th Int. Conf. Reliab. Infocom Technol. Optim. (Trends Futur. Dir., pp. 1174–1178, 2020, doi: 10.1109/ICRITO48877.2020.9197920.

[19] D. Dai et al., "Detecting, locating and crossing a door for a wide indoor surveillance robot," 2013 IEEE Int. Conf. Robot. Biomimetics, ROBIO 2013, no. December, pp. 1740–1746, 2013, doi: 10.1109/ROBIO.2013.6739719.

[20] S. Roy, T. Vo, S. Hernandez, A. Lehrmann, A. Ali, and S. Kalafatis, "IoT Security and Computation Management on a Multi-Robot System for Rescue Operations Based on a Cloud Framework," Sensors, vol. 22, no. 15, 2022, doi: 10.3390/s22155569.

[21] H. Sadek, R. Bassim, S. H. El-Ghonemy, M. Soltan, and D. El-Serafi, "Clinical characteristics and cognitive functions of late-onset psychoses: A case-control study," Middle East Curr. Psychiatry, vol. 19, no. 3, pp. 149–156, 2012, doi: 10.1097/01.XME.0000415592.81690.e8.

[22] "What is Blockchain Technology? - IBM Blockchain | IBM." https://www.ibm.com/my-en/topics/what-is-blockchain (accessed Feb. 14, 2023).

[23] "What's a blockchain?" https://www.centralcharts.com/en/gm/1-learn/1-cryptocurrency/42-trading/699-definition-of-blockchain (accessed Jul. 07, 2023).

[24] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," Proc. - 2017 IEEE 6th Int. Congr. Big Data, BigData Congr. 2017, pp. 557–564, 2017, doi: 10.1109/BigDataCongress.2017.85.

[25] S. Huh, S. Cho, and S. Kim, "Managing IoT devices using blockchain platform," Int. Conf. Adv. Commun. Technol. ICACT, pp. 464–467, 2017, doi: 10.23919/ICACT.2017.7890132.

[26] E. Androulaki et al., "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," Proc. 13th EuroSys Conf. EuroSys 2018, vol. 2018-Janua, 2018, doi: 10.1145/3190508.3190538.

[27] I. Karabegović, E. Karabegović, M. Mahmić, and E. Husak, "The application of service robots for logistics in manufacturing processes," Adv. Prod. Eng. Manag., vol. 10, no. 4, pp. 185–194, 2015, doi: 10.14743/apem2015.4.201.

[28] M. Azfar, C. K. Wei, and G. Leng, "Design and Development of an Indoor UAV," no. 1, pp. 89–93, 2015.

[29] G. Li, H. Wang, X. Ying, and J. Liu, "A proxy-based cloud infrastructure for home service robots," Proc. 2015 27th Chinese Control Decis. Conf. CCDC 2015, no. 61375087, pp. 5718–5723, 2015, doi: 10.1109/CCDC.2015.7161824.

[30] A. Koubâa et al., "Turtlebot at Office: A Service-Oriented Software Architecture for Personal Assistant Robots Using ROS," Proc. - 2016 Int. Conf. Auton. Robot Syst. Compet. ICARSC 2016, pp. 270–276, 2016, doi: 10.1109/ICARSC.2016.66.

[31] S. Muszynski, J. Stuckler, and S. Behnke, "Adjustable autonomy for mobile teleoperation of personal service robots," Proc. - IEEE Int. Work. Robot Hum. Interact. Commun., pp. 933–940, 2012, doi: 10.1109/ROMAN.2012.6343870.

[32] X. Wang, T. Zhao, and D. Wang, "Grab application of remote operation service robot," J. Phys. Conf. Ser., vol. 1633, no. 1, 2020, doi: 10.1088/1742-6596/1633/1/012031.

[33] S. Jeon and J. Lee, "Multi-robot control architecture for hospital delivery service in unstable network environment," ICINCO 2017 - Proc. 14th Int. Conf. Informatics Control. Autom. Robot., vol. 2, no. Icinco, pp. 270–277, 2017, doi: 10.5220/0006410502700277.

[34] R. Teng, S. Araki, S. Shimizu, K. Yano, and Y. Suzuki, "Multi-Channel Utilization for Local Data Sharing in Multi-Layered Wireless Robotic Networks," 2019 IEEE Int. Conf. Pervasive Comput. Commun. Work. PerCom Work. 2019, pp. 1009–1014, 2019, doi: 10.1109/PERCOMW.2019.8730666.

[35] K. Makino and T. Murase, "Cooperative Mobility Control with Longcut Route Selection to Form Ad Hoc Networks for Multiple Autonomous Mobile Robots," 2021 IEEE 10th Glob. Conf. Consum. Electron. GCCE 2021, pp. 930–931, 2021, doi: 10.1109/GCCE53005.2021.9621922.

[36] X. Dai, X. Ning, Z. Yao, and H. Shao, "Integrating cloud model in evolutionary algorithm for path planning of mobile robots," 2010 IEEE Int. Conf. Inf. Autom. ICIA 2010, pp. 2352–2356, 2010, doi: 10.1109/ICINFA.2010.5512189.

[37] B. Vogel, Y. Dong, B. Emruli, P. Davidsson, and R. Spalazzese, "What is an open IoT platform? Insights from a systematic mapping study," Futur. Internet, vol. 12, no. 4, pp. 1–19, 2020, doi: 10.3390/FI12040073.

[38] M. H. Md Saad, M. H. S. Akmar, A. S. S. Ahmad, K. Habib, A. Hussain, and A. Ayob, "Design, Development Evaluation of A Lightweight IoT Platform for Engineering Scientific Applications," 2021 IEEE 12th Control Syst. Grad. Res. Colloquium, ICSGRC 2021 - Proc., no. August, pp. 271–276, 2021, doi: 10.1109/ICSGRC53186.2021.9515199.

[39] H. Pu, L. He, P. Cheng, M. Sun, and J. Chen, "Security of Industrial Robots: Vulnerabilities, Attacks, and Mitigations," IEEE Netw., pp. 1–12, 2022, doi: 10.1109/MNET.116.2200034.

[40] S. Yu, Towards the Cloud Computing from, vol. 2. Springer International Publishing, 2018.

[41] W. Dudek and W. Szynkiewicz, "Cyber-security for mobile service robots - Challenges for cyber-physical system safety," J. Telecommun. Inf. Technol., no. 2, pp. 29–36, 2019, doi: 10.26636/JTIT.2019.131019.

[42] G. Lawitzky and M. Buss, "Service robots," IT - Inf. Technol., vol. 49, no. 4, pp. 211–212, 2007, doi: 10.1524/itit.2007.49.4.211.

[43] M. N. Zhukova, V. V. Zolotarev, V. G. Zhukov, and A. S. Polyakova, "Service robot security from unauthorized access by connection control," Proc. - Int. Conf. Dev. eSystems Eng. DeSE, vol. October-20, pp. 526–529, 2019, doi: 10.1109/DeSE.2019.00102.

[44] K. Chellappan and M. S. A. Razak, "Adapting Service Robot Mechanism in Designing Movable Makerspace for Knowledge Society Building," 2021 6th Int. Conf. Robot. Autom. Eng. ICRAE 2021, pp. 339–343, 2021, doi: 10.1109/ICRAE53653.2021.9657792.

[45] J. Wang, J. Zheng, S. Zhang, J. He, X. Liang, and S. Feng, "A face recognition system based on local binary patterns and support vector machine for home security service robot," Proc. - 2016 9th Int. Symp. Comput. Intell. Des. Isc. 2016, vol. 2, pp. 303–307, 2016, doi: 10.1109/ISCID.2016.2079.

[46] H. Cai, X. Liu, and A. Cangelosi, "Security of Cloud Intelligent Robot Based on RSA Algorithm and Digital Signature," 2019 IEEE Symp. Ser. Comput. Intell. SSCI 2019, pp. 1453–1456, 2019, doi: 10.1109/SSCI44817.2019.9002649.

[47] P. Kotuszewski et al., "Cyber-Security Assessment of Industry 4.0 Enabled Mechatronic System," Complexity, vol. 2021, 2021, doi: 10.1155/2021/6670625.

[48] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Integration of Blockchain and Cloud of Things: Architecture, Applications and Challenges," IEEE Commun. Surv. Tutorials, vol. 22, no. 4, pp. 2521–2549, 2020, doi: 10.1109/COMST.2020.3020092.

[49] A. K. Shrestha and J. Vassileva, "Towards decentralized data storage in general cloud platform for meta-products," ACM Int. Conf. Proceeding Ser., 2016, doi: 10.1145/3010089.3016029.

[50] S. Uthayashangar, T. Dhanya, S. Dharshini, and R. Gayathri, "Decentralized Blockchain Based System for Secure Data Storage in Cloud," 2021 Int. Conf. Syst. Comput. Autom. Networking, ICSCAN 2021, 2021, doi: 10.1109/ICSCAN53069.2021.9526408.

[51] X. Su, I. Ullah, M. Wang, and C. Choi, "Blockchain-Based System and Methods for Sensitive Data Transactions," IEEE Consum. Electron. Mag., vol. 2248, no. c, pp. 1–9, 2021, doi: 10.1109/MCE.2021.3076985.

[52] K. Yu, L. Tan, M. Aloqaily, H. Yang, and Y. Jararweh, "Blockchain-Enhanced Data Sharing with Traceable and Direct Revocation in IIoT," IEEE Trans. Ind. Informatics, vol. 17, no. 11, pp. 7669–7678, 2021, doi: 10.1109/TII.2021.3049141.

[53] B. Sun, Q. Dang, Y. Qiu, L. Yan, C. Du, and X. Liu, "Blockchain Privacy Data Access Control Method Based on Cloud Platform Data," Int. J. Adv. Comput. Sci. Appl., vol. 13, no. 6, pp. 10–18, 2022, doi: 10.14569/IJACSA.2022.0130602.

[54] S. M. Haris, M. Zamry, and A. Samah, "A CONCEPTUAL DESIGN OF CLOUD-BASED AUTONOMOUS GROUND VEHICLE ROBOT NAVIGATION CONTROL FOR IR4 . 0 APPLICATIONS," vol. 7, no. 28, pp. 164–175, 2022, doi: 10.35631/JISTM.728011.