Alexandria Engineering Journal (2023) 69, 275–287

HOSTED BY

ELSEVIER

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

ORIGINAL ARTICLE

The internet of things healthcare monitoring system based on MQTT protocol

Hamoud H. Alshammari

Department of Information Systems, Computer and Information Sciences College, Jouf University, Saudi Arabia

Received 24 December 2022; revised 21 January 2023; accepted 29 January 2023
Available online 9 February 2023

KEYWORDS

Message queuing telemetry transport;
Internet of things;
Healthcare system

Abstract In overpopulated nations, the need for medical treatment is increasing along with the population; healthcare difficulties are becoming more common. The population's need for high-quality care is growing despite decreasing treatment costs. Because of technological improvements, a machine may remotely monitor health, which is more reliable than manual monitoring. The time required for individualized training may be shortened, and the dependability of complicated equipment may be increased. This study recommends a real-time remote patient monitoring system based on the Internet of Things (IoT) to assure the accuracy of the vital real-time signal. The vital real-time signal is sent from the proposed method to the website using the Message Queuing Telemetry Transport (MQTT) protocol. This work aims to read and analyze patients' vital signs and reduce the latency while transmitting the signals.

1. Introduction

The healthcare sector has recently experienced considerable development and suggestively increased employment and revenue [1]. The diagnosis of diseases and other irregularities in the human body needed a physical examination in the hospital a few years ago. The majority of individuals had to remain in the hospital while receiving therapy. This increased the expense of healthcare while also placing pressure on rural and distant healthcare facilities. With the progress in technology over the years, it is now possible to diagnose numerous ailments and keep track of one's health utilizing small gadgets like smartwatches.

Additionally, technology has changed the healthcare system from being focused on hospitals to being focused on patients [2]. Blood pressure, blood glucose, pO2, and other clinical analyses, for instance, can be measured at home without the help of a healthcare proficient [3]. Clinical data may also be sent from distant areas to healthcare institutions using advanced telecommunications technology. The accessibility of medical facilities has increased thanks to such communication services and rapidly evolving technology. Machine learning, big data analytics, the Internet of Things (IoT), remote sensing, cloud applications, and cloud computing are examples of this technology [4].

The Internet of Things has expanded human interaction options with the outside world while simultaneously enhancing freedom. With cutting-edge protocols and algorithms, IoT has significantly impacted worldwide communication. Many different gadgets, wireless sensors, household appliances, and

E-mail address: hhalshammari@ju.edu.sa

Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2023.01.065
1110-0168

electrical equipment are connected to the Internet using this. The fields of agriculture, vehicles, homes and healthcare all use IoT in some capacity. The Internet of Things (IoT) is becoming increasingly popular due to its increased accuracy, reduced cost, and improved event prediction advantages.

Additionally, the advancement of computer and mobile technology, the accessibility of wireless technology, and the growth of the digital economy have all contributed to the quick IoT revolution [5]. To collect physiological data from patients' bodies, such as temperature, pressure rate, electrocardiogram (ECG), electroencephalogram (EEG), and so on, embedded or wearable sensors are utilized in healthcare applications. Environmental data can also be captured, including temperature, humidity, time, and date [6].

These data enable the development of precise and pertinent assessments of the patient's health status. The Internet of Things (IoT) system also requires the storage and accessibility of data since a significant volume of data is collected and recorded from many sources. Doctors, careers, and other authorized individuals have access to the data acquired by the above sensing devices. With cloud/server technology, healthcare professionals may quickly diagnose patients and take necessary medical action. The users, patients, and communication module work together continuously to provide efficient and secure transmission. The development of IoT systems for healthcare monitoring, management, privacy, and confidentiality are extensively covered in the literature. These successes show how useful IoT is for the healthcare industry and how promising it is going forward. The main challenge while emerging an IoT device is maintaining the quality-of-service matrix, which includes cost, availability, dependability, and information-sharing privacy [7].

1.1. IoT technologies in healthcare

Technology is essential for creating an IoT system for healthcare. This is because various technologies can improve an IoT system's functionality. Many cutting-edge technologies have been used to link diverse healthcare applications with an IoT system [8]. Three categories, namely identity technology, communication technology, and location technology, may broadly categorize this technology as shown in Fig. 1.

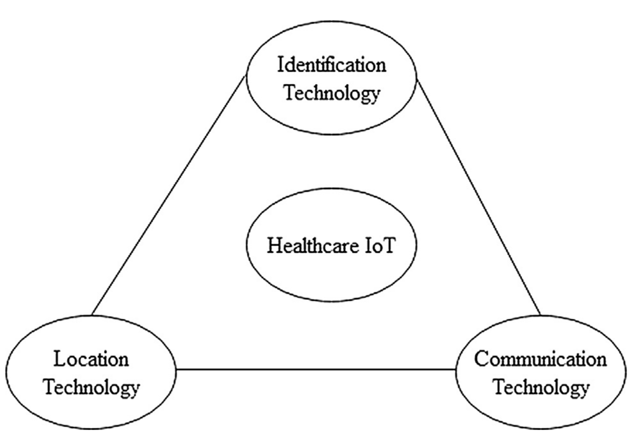


Fig. 1 Healthcare IoT technology classification.

1.2. Identification technology

Each authorized organization is given a distinctive identity as part of the identification process, which enables quick identification and transparent data sharing [9]. Every resource connected to the healthcare system, in general, comes with a unique digital identifier. This guarantees the link between the resources in a digital domain and their identification. Numerous identification criteria have been documented in the literature.

The Open Software Foundation has created two unique identifiers: one globally unique and the other universally unique. A component of the Distributed Computing Environment (DCE), the universally unique identity, may be used without the need for centralized coordination. The sensors and actuators in a healthcare network are recognized and addressed independently, aiding in the system's smooth operation. However, because of the ongoing development of IoT-based technologies, a component's distinctive identity may change during the IoT system's life cycle [10]. Therefore, to maintain its integrity, the system or device needs a mechanism to update this information. This might be explained by the possibility that changing the settings could impact the tracking and diagnosis of the network device.

1.3. Communication technology

The linking of the healthcare IoT network is made possible by communication technologies. These may be broadly categorized into medium-range and short-range communication technologies. In contrast to medium-range communication technologies, which typically support communication over long distances, such as communication between a base station and the central node of a body area network, short-range communication technologies are the protocols used to connect objects within a constrained range or a BAN (BAN). In the case of short-range communication, the communication distance might range from a few millimeters to several meters. In IoT applications, short-range communication techniques are most frequently utilized. Bluetooth, RFID, Wi-Fi, Zigbee, and other popular communication technologies.

1.3.1. Radio-frequency identification (RFID)

Face-to-face communication is facilitated by RFID (10 cm–2 00 m). A reader and a tag make up the device. A microcontroller and an antenna are used to make the tag. In the IoT ecosystem, it is used to precisely identify a product or device (such as medical equipment). The reader makes a radio wave connection with a tag to communicate or receive data from the item. In the Internet of Things, an electronic product code serves as the data format for the tag (EPC). RFID enables medical practitioners to locate and keep track of medical equipment quickly. The main advantage of RFID is that it doesn't need an external power source. It is a somewhat uncertain protocol, though, and connecting with a smartphone may result in compatibility concerns.

1.3.2. Bluetooth

Bluetooth and short-range wireless communication technologies use ultra-high frequency (UHF) radio waves. This expertise agrees with the wireless connection of two or more medical equipment. 2.4 GHz is the Bluetooth frequency band. The Bluetooth protocol has a 100-meter communication range. Bluetooth offers both encryption and authentication for data security. The low cost and energy efficiency of Bluetooth are its main advantages. Additionally, it assurances less interference during data transfer between the linked devices. However, this technology falls short of the requirement when the healthcare application calls for long-distance communication.

1.3.3. Zigbee

One of the common protocols used to link up, and exchange data amongst medical devices is called Zigbee. Zigbee operates in frequencies akin to Bluetooth (2.4 GHz). However, compared to Bluetooth-enabled devices, it offers a wider connection range. The network of this technology features a mesh topology. End nodes, routers, and a dispensation center make up this system. The processing center carries out data gathering and analysis. Even when one or more devices malfunction, the mesh network guarantees constant communication between other gadgets. Zigbee has several benefits, including low power feasting, fast transmission rates, and an extensive capacity network.

1.3.4. Near-field communication (NFC).

The core concept underpinning NFC is electromagnetic induction among two loop antennas positioned close to one another. This technology transmits data through electromagnetic induction, similar to RFID tags. Active and passive are the two operational modes for NFC devices. Only one device creates the radiofrequency while in passive mode; the other acts as a receiver. Without pairing, both devices can simultaneously send data when in active mode and emit RF [11]. The ease of use and effective wireless communication network of NFC are its key benefits. However, it only works for highly close-range communication.

1.3.5. Wi-Fi

The IEEE 802.11 standard is followed by Wireless Fidelity (Wi-Fi), a wireless local area network (WLAN). It has a communication range of up to 70 feet, which is more than Bluetooth's. Fast and simple network creation is possible with Wi-Fi. As a result, it is mainly used in hospitals. Wi-Fi is popular because it is easily compatible with smartphones and can offer great security and control. However, its power usage is disproportionately higher, and the network is chaotic.

1.3.6. Satellite

In remote and geographically distant locations, such as rural regions, highlands, heaps, oceans, and so forth, where other technologies for communication are difficult to reach, satellite communication has been proven to be more beneficial and effective. The satellite gathers signals from the ground and amplifies them before transmitting them back to Earth. More than 2000 satellites are in orbit around the planet. Satellite communication has several advantages, including quick internet access, dependability, and technology compatibility. However, compared to other communication methods, satellite transmission has a very high-power need.

1.4. Location technology

The healthcare network uses real-time location systems (RTLS) or other location technologies to identify and locate objects. Based on how the resources are allocated, it also monitors how the therapy is progressing. The GPS, or global positioning system, is one of the most frequently utilized technologies. Satellites are used for tracking reasons. As long as there is a direct line of sight between the target and the four satellites, GPS can locate the item. It might be used in IoT for healthcare to find an ambulance, a doctor, a caregiver, a patient, etc.

The use of GPS, however, is restricted to outdoor settings since nearby structures may interfere with an object's ability to communicate with a satellite. A local positioning system (LPS) network can be utilized successfully in such circumstances. By detecting the radio signal that the moving item emits to a variety of already placed receivers, LPS can follow that. LPS may also be used with several short-range communication technologies, including RFID, Wi-Fi, Zigbee, and others [12]. However, due to its benefit of having a better temporal resolution, ultra-wideband (UWB) radio is recommended. The receiver can then precisely calculate the arrival time. It uses a UWB-based localization technology that tracks motion based on the time difference of arrival (TDOA).

Real-time tracking, patient data management, medical emergency managerial staff, and blood data management are just a few medical applications using IoT that are now being used in the healthcare system [13]. The Internet of Things is used in the healthcare industry to connect various medical resources and provide every-one who needs them with effective and efficient care. Applications like telemedicine, telehealth, and personal health records employ existing IoT-DHSs models. Distributed systems consist of several independent computers that seem to consumers as a single, unified computing system. For high-performance computing tasks, distributed systems are used [14]. The main goals of distributed systems are for users or programs to have simple access to remote resources. Grid computing systems are an aggregation of workstations or computers that all run a different administrative domain, hardware, software, and established network topology, as opposed to cluster computing systems, which are made up of machines that all run the same operating system.

Dispersed information systems are supported by transaction processing and business application systems. The main aim of the distributed health monitoring system is to develop the tendency and integrated telemedicine, telehealth, PHR, etc. [15]. The distributed healthcare monitoring system addresses some of the shortcomings of the current IoT design, which is excessively centralized and extensively reliant on the back-end core network for all decision-making progressions [16]. In healthcare contexts, monitoring systems offer highly accurate actor and equipment tracking. Clinical staff actions are observed to guarantee high-quality healthcare. Systems for monitoring patients' health indicators are also employed. They provide real-time analysis and feedback regarding clinical staff and patients in emergency scenarios. Contribute to reducing the number of fatalities brought on by negligence or human error. Computer vision, real-time location systems, and combined CV-RTLS are three cutting-edge techniques for monitoring patients and healthcare personnel in healthcare systems [17].

1.5. Objectives

- The research aims to reduce the latency while transmitting the signals.
- To pursue system development in IoT healthcare applications.
- To state the importance of the IoT in the health care system.

1.6. Challenges

The nodes in the system must create a large amount of data and sample sensors quickly. For this, the batteries' high energy is required.

The human body affects wireless broadcast signal broadcast. Signal transmission is impacted by diffraction around the body and reflection from the body.

The healthcare system uses several nodes with various signals to continuously produce a large amount of streaming data. Data collisions and high energy usage result from this.

Systems in the healthcare industry should offer real-time communication with assured latency to give patient feedback in the event of emergencies like patient falls.

Healthcare system sensors are closely tied to one another. The detected signals are utilized to produce or recreate the initial motions.

Scalability is required for IoT-based healthcare systems' networks, services, apps, and back-end databases.

Because healthcare systems deal with a multi-patient setting, it is essential to correctly identify patients inside the system.

The paper is classified as follows: Section 2 reviewed some of the works done on the same subject. Section 3 provided information regarding the methodology. Section 4 provided the discussion, results, and the comprehensive progress of the suggested approach to current best practices. The last section is 5, where the paper's conclusion.

2. Related work

A WLAN-based end-to-end flow remote health monitoring system has been introduced. The system uses medical sensors to gather the patient's vital signs, such as heart rate, blood pressure, pulse oximetry, and body temperature. The work in this study did not incorporate real-time ECG signals. The primary controller for processing and analyzing the data is an Arduino Yun. The application programming interface (API) is then used to upload the data through WLAN to the cloud. Doctors can use the internet server to verify the patient's medical information and to make remarks to the patient. Data will be momentarily retained on the memory card if the device loses its WLAN connection. Once the assembly is restored, the system will synchronize the data with the server [18].

A remote health monitoring system based on smartphones is suggested. A smartphone application is used in this system. To discuss the course of therapy, the persistent can use a smartphone to video conference with the doctor. This technology secures the patient's video data with Skype's AES-style encryption. This system does not support real-time mode. The patient must manually submit the vital signs obtained to the server [19].

Using medical data on the internet and through mobile application platforms is made possible by providing an RF-based remote monitoring system. The entire system comprises a coordinator node, website and database server, edge devices, and graphical user interface (GUI) element. The primary server receives the data that has been collected. The GUI allows users to view the data and analyze the outcomes. The sluggish server response time and the transmission latency preclude this system from running in real-time [20].

Doctors may remotely check on their patients' health problems using an internet-based IoT-based healthcare system, which has been demonstrated. The biometric data was collected and wirelessly sent to the internet server using this module's E-Health Sensor Shield Kit. The microcontroller temporarily stored the sensor data before transferring it to the web database. The user must log in using a unique ID to view the data. The system needs around 15 s to complete the process of gathering health data, sending it to the web server, and visualizing it on the webpage. With the proliferation of various portable devices designed to enable remotely mindful, the trend of remotely monitoring patients' health from their homes is growing. By exchanging health information with human services organizations like experts, medical attendants, and professionals, the cloud, IoT, and mobile technologies make monitoring the patient's well-being easier. We have provided the emergency clinic board with a medical services framework that enables watchmen and specialists to remotely check on patients' welfare online. The essential core focus is on remote observation and direction awareness through data transmission in a verified manner. People require ongoing medical treatment administrations to be completed [21].

Edge Cloud and Web Real-Time Communication (WebRTC) is suggested for a patient remote monitoring system. In addition to serving as storage, Edge Cloud also serves as a platform for communication between patients and remote patient monitoring systems. Push and pull are the two operating modes for the system. When push-mode remote monitoring is used, the system notifies the user if any suspicious health data is found. On the other hand, pull-mode remote tracking enables the user to evaluate previously obtained data. WebRTC supports a variety of channels for communication between the cloud server and the central system. The system's analytical engine will examine the data and video gathered. Instead of vital sign data, this system primarily focuses on optimizing and sending visual data [22].

A Raspberry Pi is used to construct an IoT-based e-health system. The data in a particular chart was recorded using a server-side Node program, providing a web interface for the user to see the chart. The e-health system is furnished with temperature, galvanic skin response, airflow, pulse oximeter, and blood pressure sensors. With this technology, the user may choose between a live session and a recorded session for data recording. The data will only be momentarily saved in the live assembly mode. The information will be saved for the recording session forever [23].

An IoT device called the remote patient well-being observation framework might be used at home by elderly people or patients to monitor their continuous health measurements, including temperature, blood pressure, and electrocardiograms. This IoT device would then alert the users if a crisis materialized, which would fluctuate the sensor readings outside the typical range. A thermometer, ECG sensor, and sphygmomanometer are used to build this gadget, and an Arduino is used to transfer data to servers using WIFI. The servers decide on the information shown on adjacent handheld devices. In the unlikely event that the sensor readings are outside the normal range, a server alert will be issued to the client [24].

As people use sensor-based well-being monitoring devices, problems start to appear. Even though health monitoring devices are small, seeing them confuses the hardware and their application. Sensors may be used to detect a variety of modalities, including the pulse and cardiac rate, body temperature, and pulse, and the information can then be stored for further review and analysis. Although there is a chance at every perspective to screen, certain misunderstandings need to be clarified. There aren't many unanswered questions, such as helping blind people alone, identifying and sending a crisis response team to the area using GPS, identifying and storing high and low weight in a database, initiating the use of an emergency treatment child by a person nearby, and encouraging people to help the patient. The information that has been collected and stored will help the patients use it for future use. The method of determining the problem is looked at, and a philosophy is suggested. A WLAN-based system was demonstrated for longer-term health monitoring with a stream [25]. The system uses therapeutic sensors to gather silent vital signs such as body temperature, pulse oximetry, blood pressure, and heart rate.

An ECG flag in real time was not required for the assignment. The most effective regulator for handling and analyzing data is Arduino Yun. At that time, the data is sent via WLAN using the API into the cloud. The Online Patient Check Server allows specialists to login in, access clinical data, and refer comments to persistent server users. The data will be briefly kept in the memory flash in the unlikely event that the device loses its WLAN connection [26]. The framework will organize the data from the server once the connection to it has been established again.

The usage of web and mobile application platforms to get information about the user's medical treatment is illustrated by utilizing an RF-based further healthcare monitoring structure. The framework comprises radar hubs, controller nodes, and client, web, and file server interfaces [27]. The information is gathered via the sensor hub. The collected data is moved to a central server. Clients can take notes and review the outcomes via the GUI. The system can't support real-time due to broadcast interruption and slow server response. By using WEBRTC in addition to Edge Cloud, a monitoring system that is inaccessible to the public is created. That isn't the function, but it is also used as a communication channel between the patients and a more advanced silent checking system [28]. The framework offers two distinct operating modes: push and pull. If aberrant health information is identified in remote monitoring push mode, the framework notifies the client by sending an alarm. Pull mode is more focused on getting the client to review the already obtained data. Many of the ways the cloud server interacts with the majority of frameworks are supported by WebRTC. The system's explanatory motor analyses the data that has been gathered and the video [29].

3. Methodology

The four components of the proposed patient monitoring system are shown in Fig. 2, a set of vital sign sensors, a microcontroller, an MQTT broker that serves as a communication module, and a receptor (Mobile application or website). Vital signs, including blood pressure, heart rate, and body temperature, will all be recorded. Then these data are collected using various types of sensors. The microcontroller (Arduino or Raspberry Pi) will send the collected data to (MQTT) broker via Wi-Fi___33. Finally, the MQTT broker publishes these readings data to the web server or Mobil application.

An AD8232 sensor is used to gather data from the heart patient. It locates and amplifies the critical signal using the electrical activity of the heart muscle as a starting point. The system's primary controller is an Arduino ESP 32 with Wi-Fi___33. Before using a Wi-Fi___33 module to transmit real-time data to an MQTT broker, the primary controller will review the information it has already collected. An MQTT (Message Queuing Telemetry Transport) broker housed on a Raspberry Pi 3 transmits the data to the website as shown in Fig. 3.

This experiment makes use of an MQTT relay. The JSON data is delayed before being stored in the database since it subscribes to the same Topic as the webserver in the MQTT broker. The recommended method may, therefore, simultaneously show the vital real-time signal on a web server and save it in a database. In addition to the website, the doctor may access an Android mobile application to view the current vital signs (App). Users may observe a real-time ECG chart and pulse rate with the program (BPM). The BPM, which is updated every-three seconds, is calculated using the real-time data that has been obtained. Doctors and nurses can see the stored historical data by connecting to the database server.

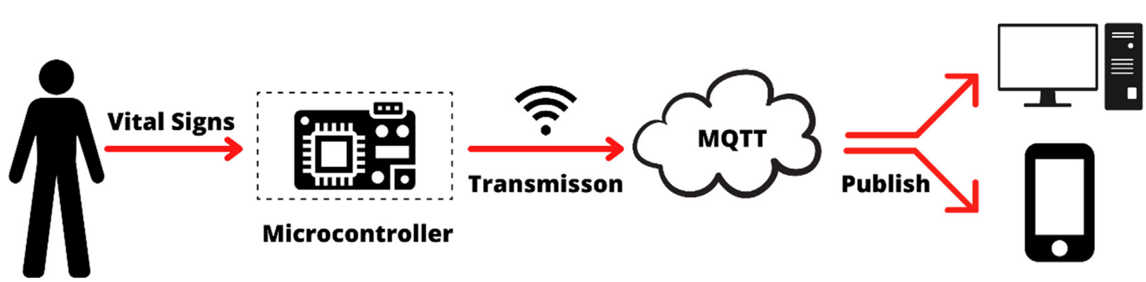


Fig. 2 Components of the proposed system.

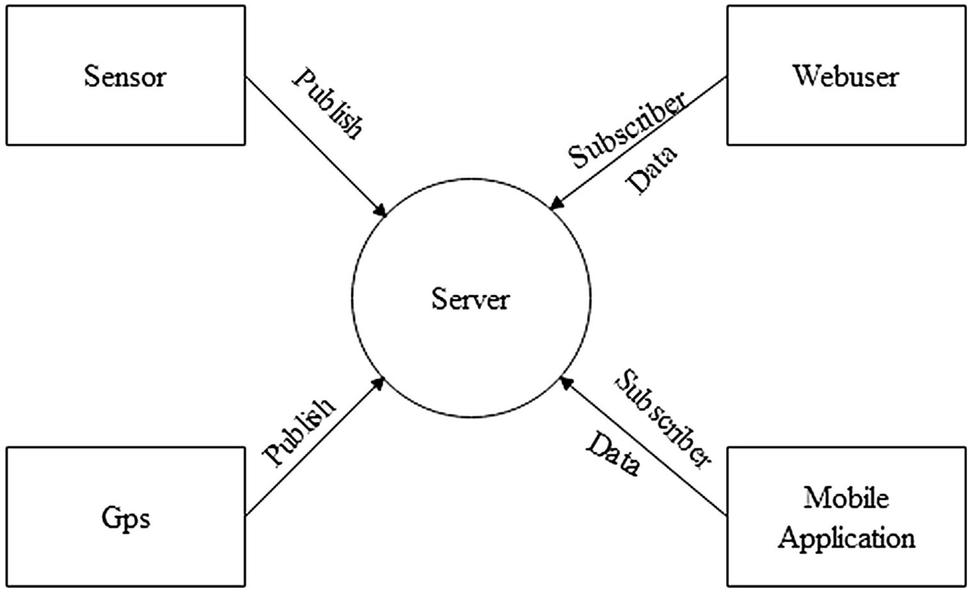


Fig. 3 MQTT network Communication.

Instead of using the Hypertext Transfer Protocol (HTTP), this work employs the MQTT network protocol since it uses less bandwidth and electricity and is more suited for Internet of Things (IoT) development. Before transferring the data to the website and database, the crucial signals data is condensed into a JSON Format string. The efficiency of data conversion on the client side is much improved since both the web server and the database have internal functions that can accept JSON-structured data.

JavaScript programming language was used to create the web server. A vital real-time signal is shown via web sockets based on the JSON data sent by the projected healthcare monitoring system. Additionally, the webserver calculates suite loss and jitter delay.

The equation for calculating the package loss is given as follows (1)

$$P_L = N_S - N_R \tag{1}$$

NS stands for the number of packages delivered by the central controller, NS for the number of packages received by the web server, and P_L stands for package loss. The jitter delay is calculated using equation (2).

$$P_L = N_S - N_R \tag{2}$$

jitter delay location, the time difference between packages delivered by the primary controller and those received by the web server, is denoted by the abbreviation TS and T_R, respectively. Using the equation, the average jitter delay may be calculated.

$$P_L = N_S - N_R \tag{3}$$

where S is the total jitter delay and is the total number of packages.

3.1. MQTT protocol

The most well-known protocol in internet history is HTTP (Hypertext transfer protocol). The HTTP protocol has become extremely important for data transport in this diverse internet system. The amount and diversity of IoT devices compared to internet devices set the internet of things apart from the internet. Due to restrictions like header size, latency, completely connection-oriented design, and other issues, effectively connecting IoT devices and the internet might not be as simple as it would seem. The MQTT Protocol is explored, and its numerous uses are to condense the subject matter.

From the MQTT protocol, IBM developed it. The communication model is where HTTP and MQTT fundamentally diverge. While MQTT uses the Publish-Subscribe architecture, HTTP uses the Request-Response communication mechanism. Sensors, dashboards, and mobile apps are included in generic internet of things deployments. Data acquisition involves multiple switches and routers. The publisher is the first of three main entities engaged in the MQTT protocol. The publisher gathers information from various sources, including embedded mobile sensors, deployed sensors in machines or wearables, and more. Subscription is the second entity. The subject the subscriber selects for which the publisher makes data available. A user dashboard or a mobile application might be the subscriber. The broker is the third and most significant entity. The primary responsibility of a broker is to obtain data from the publisher and give it to the subscriber. The broker will be set up on creative and resourceful equipment that can handle many difficulties simultaneously. Most apps utilize brokers to store data in cloud services, which users access, making it simple to manage data.

When a little code footprint is necessary or network bandwidth is inadequate, a lightweight publishing and subscribing messaging protocol called MQTT was developed to be used on top of the TCP/IP protocol for communications to distant locations. One of the components required for every Publish/Subscribe messaging design is a message broker. Fig. 4 depicts the signal flow diagram for MQTT, which comprises a message broker, publisher client, and subscriber client. A client must connect to the broker to exchange messages using MQTT's publisher/subscriber mechanism.

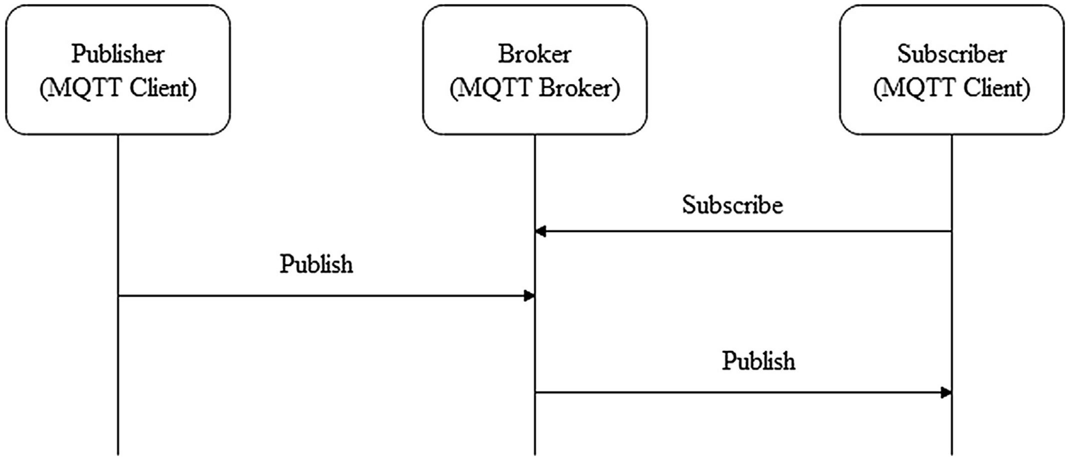


Fig. 4 Flow of signal in MQTT.

A client must subscribe to the broker's communications and be interested in the topics to hear the subjects. The client must keep the session alive by sending a keep-alive message to the broker if it is not interested in sending or receiving messages. A session between a client and broker is terminated if a broker does not receive a keep-alive message. One broker server and the Publisher and Subscriber types of clients make up MQTT. For the intriguing topic, the broker server serves as a middleman for communications between the publishing client and subscribe client. The Subscribe client chooses the subjects it considers when the Publish client offers a topic and transmits a message to the Broker server. The content of an MQTT message is primarily made up of three components: a fixed header, a variable header, and a payload. Either every order communication must contain a fixed header for 2 bytes. The format of the fixed header is specified in Table 1.

- There are 14 different message types.
- QoS Level: The three quality service levels are QoS 0, QoS 1, and QoS 2. The larger the influence of communication efficiency, the higher the degree and operational overhead.
- Length Left: in addition to the outside fixed head
- The maximal transmission size, such as the variable header and payload portion, can be extended to 4 bytes, or 256 MB.

In QoS Level 0, a message is only transmitted once by the message distribution flow; it is not checked to see if it has reached its intended recipient. As a result, it is conceivable that the message will be for messages of a certain size.

Using the status check message PUBACK, QoS Level 1 transmits the message at least once and verifies that it has been delivered. Since the server lacks proof that the message was successfully sent when PUBACK was lost, the identical message would likely be sent repeatedly. Various themes, with the corresponding payload differing in content Table 2 displays the payload.

- Sender: 64-byte source and destination data that may be empty
- TS: send the date stamp, which must be 4 bytes long and not empty
- Payload type: 1-byte form
- Content: JSON-formatted push message data.

The message is passed precisely once by QoS Level 2 using the 4-way handshake. This level virtually eliminates the likelihood of a message loss. The challenging 4-way handshake usage procedure causes the end-to-end latency in QoS Level 2. As was already noted, the MQTT protocol includes advantages such as reduced overhead, support for three levels of QoS, many-to-many connections, etc. The downside of MQTT is that it requires session maintenance for message transport. This is due to the connected gadget never having a chance to relax as it must constantly maintain a connection. The provision of different services is constrained by including a broker in the MQTT framework.

Table 2 Payload Format.

70	68	64	0
Content	Kind	TS	Sender

Table 1 MQTT Message Format.

Bit	7	6	5	4	3	2	1	0
Byte1	Type of Message				DUP Flag	QoS Level		Recollect
Byte2	Remaining Length							

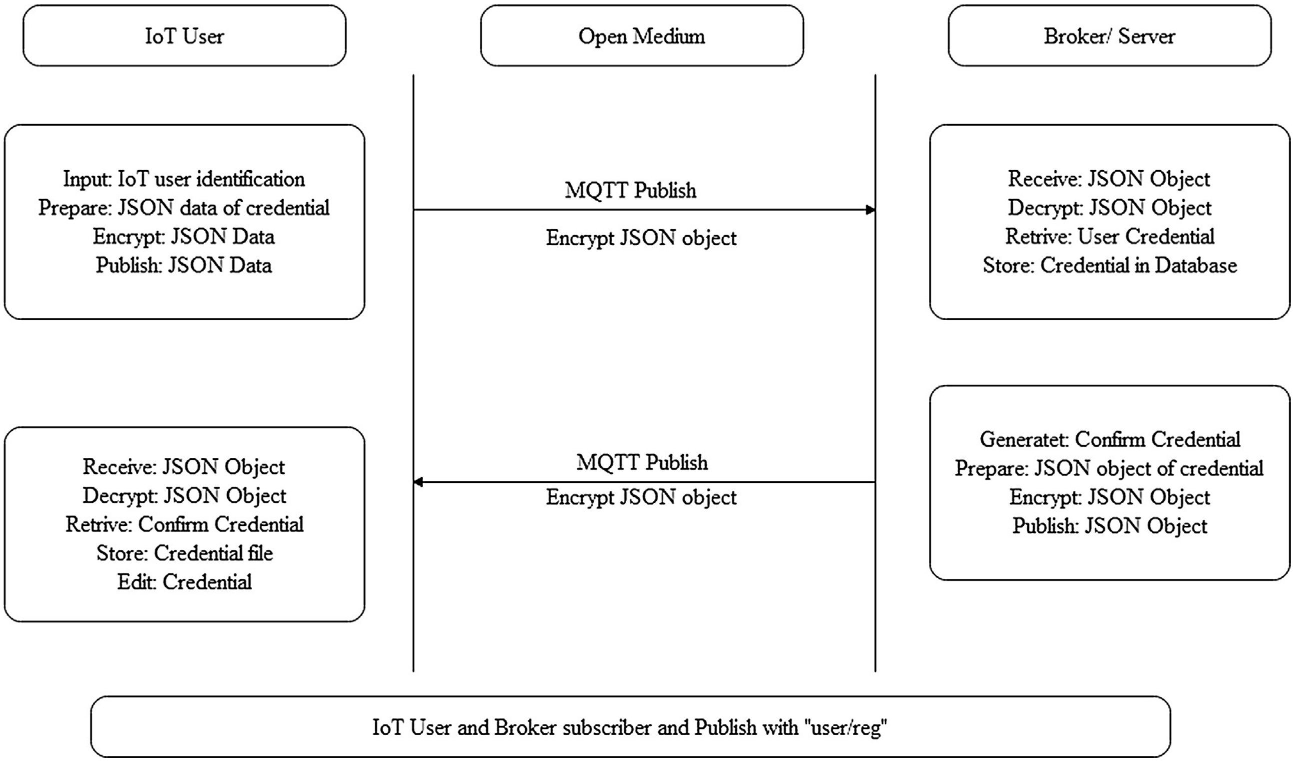


Fig. 5 User Registration Phase.

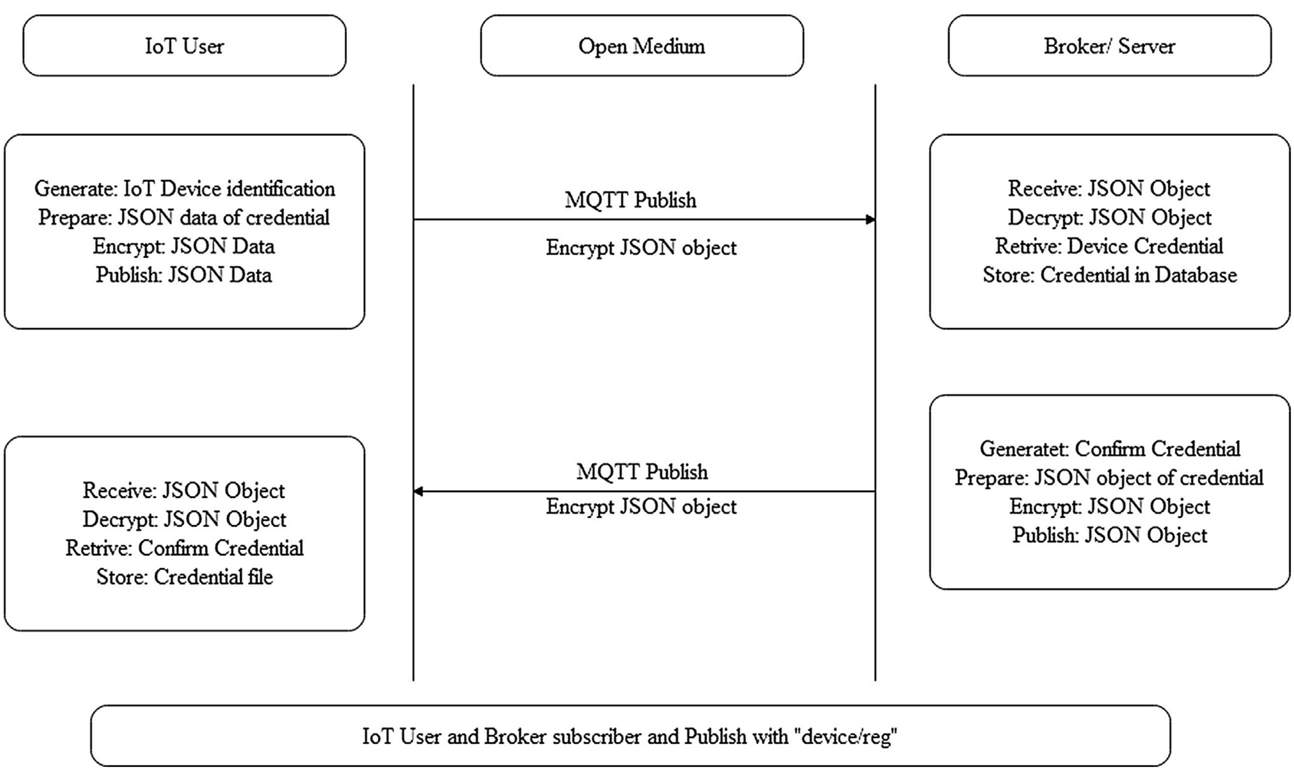


Fig. 6 Device Registration Phase.

3.2. Framework MQTT authentication

There are three steps to the suggested framework. User registration, device registration, and user authentication are among the phases. As previously mentioned, three entities are involved in MQTT communication. A broker, a subscriber, and a publisher make up the organization. Assume that the user needs access to the data from the perception device under the described paradigm. To do that, the user device must first register with the broker device, after which it must validate with the broker device to access the sensor data.

3.2.1. Phase of user registration

As seen in Fig. 5, the user interacts with the broker at this step to register using their user information.

With the broker device's pre-shared public key, the user encrypts a message created using their user information. The user uses the open channel to send this message to the broker. The private key is attached to the user information by the broker to obtain them. The broker keeps the password in the database. The broker generates any additional credentials needed by the user to validate them. The broker creates a credential object, then broadcasts it on the transmission medium after encryption. The user obtains the broker's information, confirms the broker's legitimacy, and stores the login information in a file. The user's computing device can also store data in the file with the appropriate resources.

3.2.2. Phase of device registration

As seen in Fig. 6, the device communicates with the broker to register using the device credentials at this step.

A communication that the sensing device generates using the sensing device credentials is encrypted using the pre-shared public key of the broker device. The perception device uses an open channel to transmit this message to the broker. Using the secluded key, the broker embellishes the credentials for the sensing equipment. The broker stores the credential in the database. The broker creates any extra credentials required by a sensing device to confirm authenticity. The broker creates an access control object, encrypts it, and then publishes it on the transmission medium. The sensing device obtains the broker-generated information, verifies the broker, and saves the login information to a file.

3.2.3. Phase of user authentication

This stage involves the user authenticating themselves with the broker, as seen in Fig. 7.

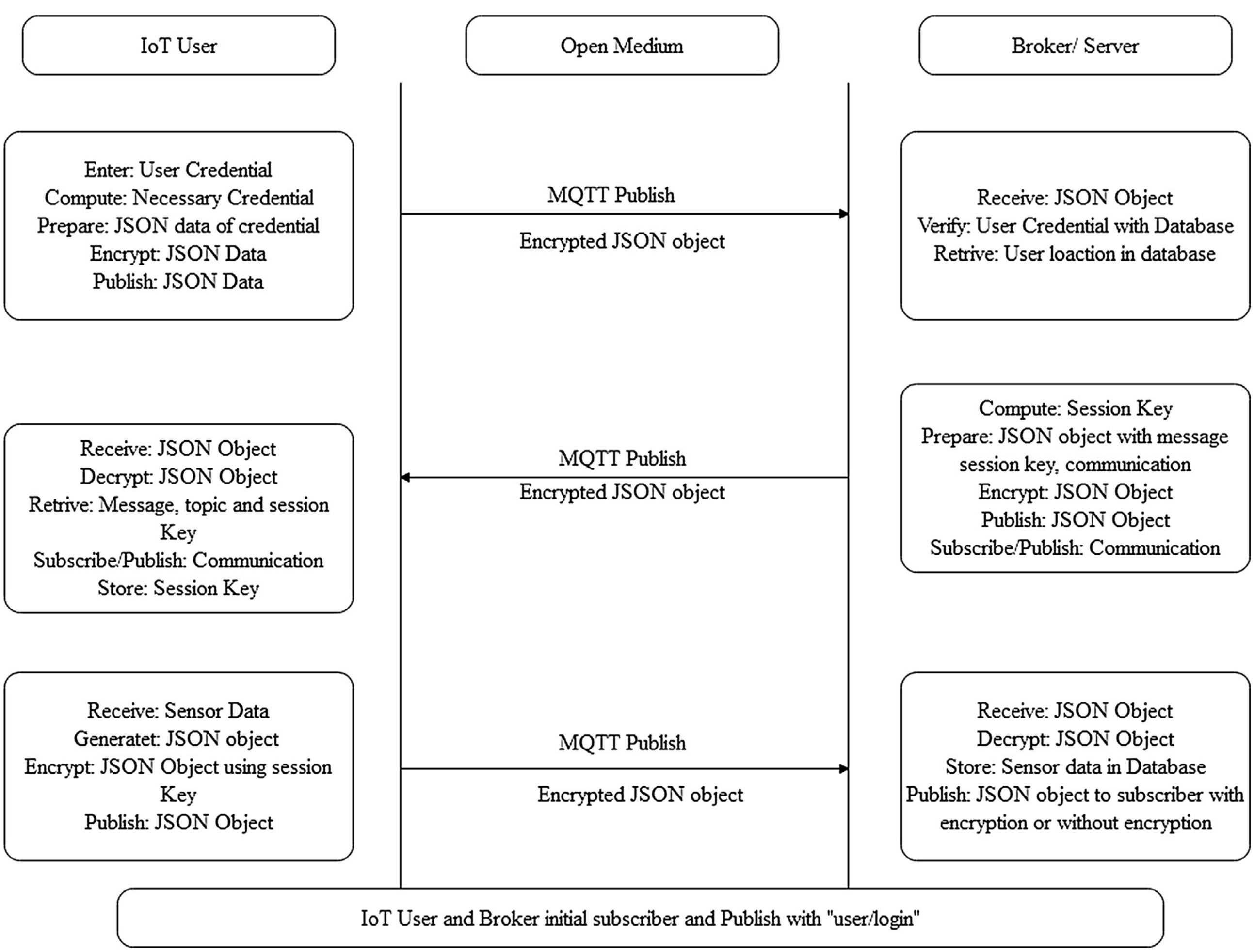


Fig. 7 User Authentication.

After receiving the user's credentials, the user device calculates the required authorizations. The computed credentials are encrypted in a JSON object created by the user's device. The user device distributes this JSON object to the broker. The broker device obtains the user's credentials from the database, where they are then verified there. The broker provides the user with the proper subjects so they may get the data from the sensing devices. For the user to securely connect with the sensing device, the broker also computes any session key credentials that may be required. The authorized message, assembly key, and communication are all included in a JSON object that the broker produces and encrypts using the user device's public key. The broker publishes this JSON item to the user's device.

The broker securely exchanges session keys and user identities with the sensing devices. The user receives the session key and the appropriate subject to contribute to each sensor. The user keeps the session key and subscribes to the specified subjects. The sensor begins to send its session key-encrypted data. Such data is encrypted by the user and stored in a broker file or a local file via the free channel.

4. Results and discussion

Consider observing the MQTT publish/subscribe messaging technology to assess the suggested methods. The execution of this performance test demonstrated the importance of the suggested messaging system's essential qualities, including its weight property and dependability-providing capabilities. The loss rates of 5, 10, 15, 20, 25, and 30 % were added and examined as appropriate. Table 3 shows the packet loss rate during the round-trip time at two different protocols, QoS1 and QoS. The MQTT protocol provides QoS 0, 1, and 2 in dependability. The typical RTT (Round Trip Time) for a single transaction is shown in Fig. 8. MQTT QoS 1 and 2 are appropriate here. These findings have nothing to do with how UDP and TCP function differently. As a result, the RTT of MQTT is shorter overall.

Table 4 and Fig. 9 show the packets required to transfer the measured biomedical message from the healthcare measurement device to the message broker, immediately sent to the management and the treatment device. In this case, this quantity was ignored. To provide a fair comparison, two different MQTT messages with the same biological message were delivered. The broker and clients for each protocol were set up on the same desktop computer and Raspberry Pi to do this.

Table 5, 6 and Fig. 10 illustrate the computations required to encrypt and decrypt sensor data. The sensor performs one addition and one multiplication throughout the encryption process. During the decryption procedure at the receiver side, the server must do one multiplication and one subtraction to get the original message from the encrypted text. The amount of time needed for encryption and decryption grows together with the message length.

Table 3 Packet Loss Rate at Round Trip Time.

Packet Loss Rate	Round Trip Time	
	QoS1	QoS
0 %	0	0
5 %	0.15	0.2
10 %	0.18	0.25
15 %	0.2	0.4
20 %	0.25	0.6
25 %	0.6	1
30 %	1	1.25

Table 4 Packets required for data transfer.

Number of Packets	Number of Messages	
	MQTT-PUB	MQTT-SUB
2	2	7
4	2	4
6	2	2
8	2	7

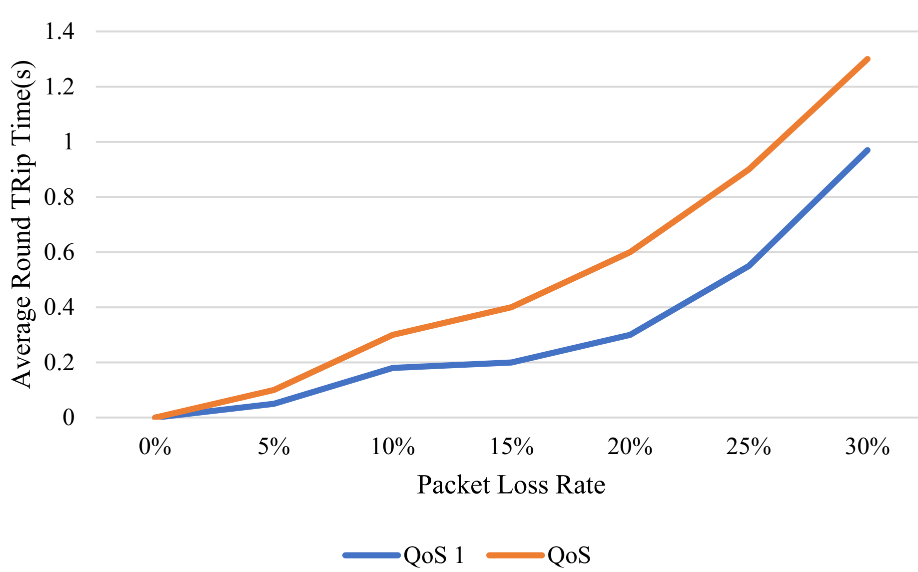


Fig. 8 Packet loss rate with round trip time.

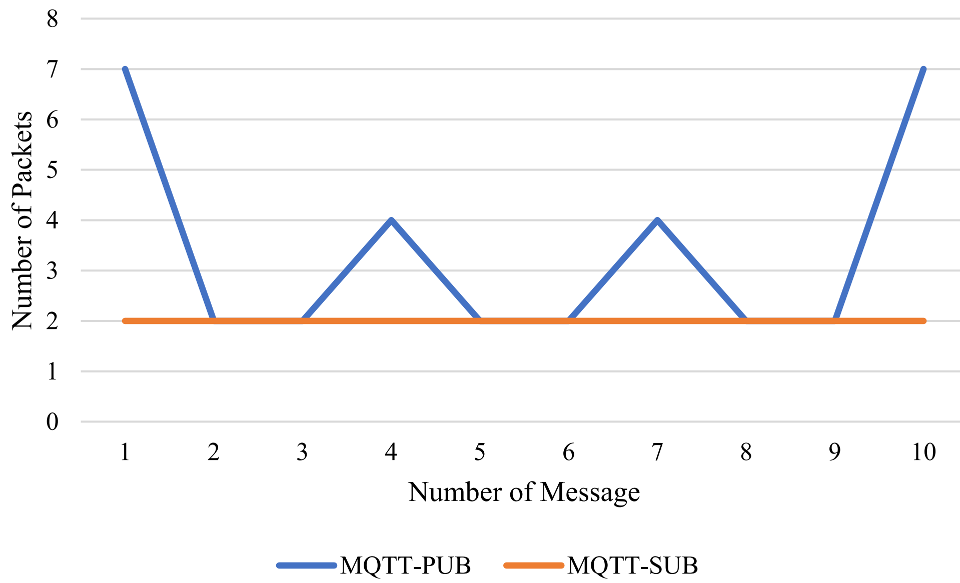


Fig. 9 Packets required for data transfer.

Table 5 Encryption Time.

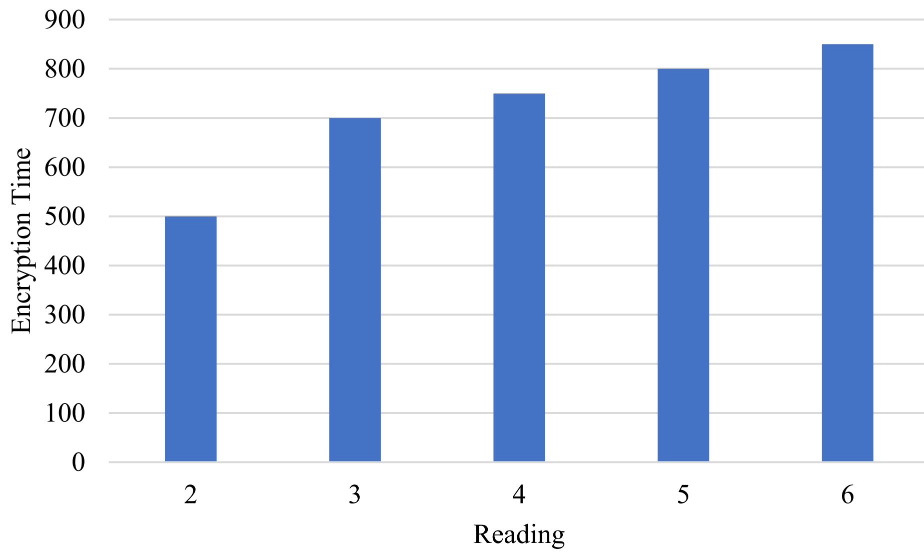
Reading	Encryption Time
2	500
3	700
4	750
5	800
6	870

Table 6 Decryption Time.

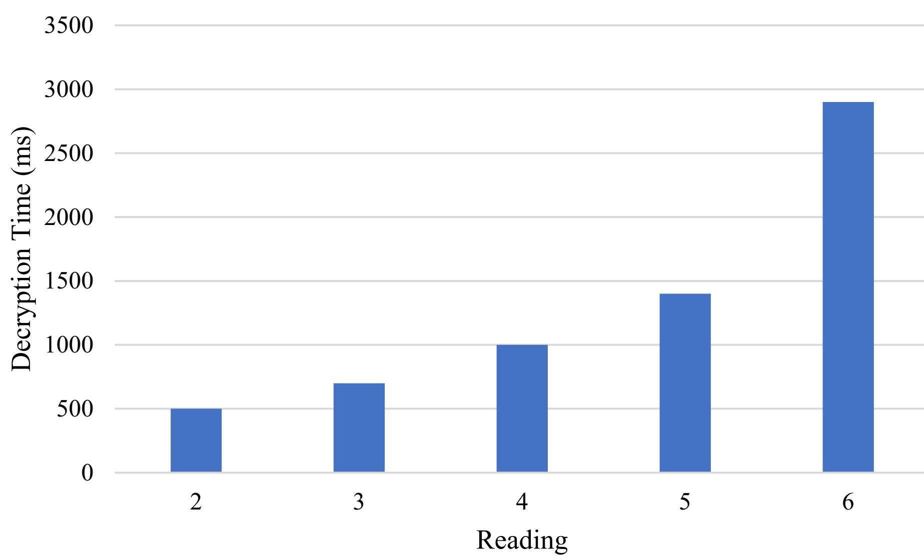
Reading	Decryption Time
2	500
3	850
4	1000
5	1495
6	2800

5. Conclusion

The proposed real-time persistent monitoring structure has been evaluated using healthcare IoT. Successful online presentation and storage of the crucial real-time signal in a cloud server. The most important protocol for IoT application layer communication is MQTT. All main sectors are currently faced with a huge difficulty related to security and privacy; hence in this paper, different security such as access control and authentication, has been emphasized. The trial results show no package losses or errors during data package transfer. The recommended system is usable by the hospital. The doctors or nurses may use a computer or smartphone to check any patient's vital signs anytime, anywhere, without going to the wards. This method can save travel time and costs, particularly for patients from residential or rural areas. The recommended strategy has the potential to improve healthcare across the nation. The proposed system's prototype was developed and implemented, and the MQTT-based system's performance was assessed regarding round-trip time and the number of transaction packets.



(a) Encryption Time



(b) Decryption Time

Fig. 10 Computation time required for Encryption and Decryption of sensor data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was funded by the Prince Nawaf bin Abdelaziz Chair for Sustainable Development in collaboration with the Deanship of Scientific Research at Jouf University under grant No (DSR2021-Prince Nawaf bin Abdulaziz Chair-11).

References

[1] Z. Ali, M.S. Hossain, G. Muhammad, A.K. Sangaiah, An intelligent healthcare system for detection and classification to discriminate vocal fold disorders, Futur. Gener. Comput. Syst. 85 (2018) 19–28.
[2] G. Yang et al, A health-IoT platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box, IEEE Trans. Ind. Inf. 10 (4) (2014) 2180–2191.
[3] M. Khan, K. Han, S. Karthik, Designing smart control systems based on internet of things and big data analytics, Wirel. Pers. Commun. 99 (4) (2018) 1683–1697.
[4] El-Komy, Amir, Osama R. Shahin, Rasha M. Abd El-Aziz, Ahmed I. Taloba, Integration of computer vision and natural

language processing in multimedia robotics application, Inform. Sci. Lett. 7(6) (2022).

[5] V. Jagadeeswari, V. Subramaniyaswamy, R. Logesh, V. Vijayakumar, A study on medical Internet of Things and Big Data in personalized healthcare system, Health Inform. Sci. Syst. 6 (1) (2018) 1–20.

[6] Abd El-Aziz, Rasha M., Ahmed I. Taloba, Fahad A. Alghamdi, Quantum Computing Optimization Technique for IoT Platform using Modified Deep Residual Approach, Alexandria Eng. J. 61 (12) (2022) 12497-12509.

[7] A. Gatouillat, Y. Badr, B. Massot, E. Sejdić, Internet of medical things: A review of recent contributions dealing with cyber-physical systems in medicine, IEEE Internet Things J. 5 (5) (2018) 3810–3822.

[8] Y. Yuehong, Y. Zeng, X. Chen, Y. Fan, The internet of things in healthcare: An overview, J. Ind. Inf. Integr. 1 (2016) 3–13.

[9] G. Shanmugasundaram, G. Sankarikaarguzhali, "An investigation on IoT healthcare analytics", International Journal of Information Engineering and Electronic, Business 9 (2) (2017) 11.

[10] H. Aftab, K. Gilani, J. Lee, L. Nkenyereye, S. Jeong, J. Song, Analysis of identifiers in IoT platforms, Digital Commun. Networks 6 (3) (2020) 333–340.

[11] G. Cerruela García, I. Luque Ruiz, M.Á. Gómez-Nieto, State of the art, trends and future of bluetooth low energy, near field communication and visible light communication in the development of smart cities, Sensors 16 (11) (1968, 2016.) pp.

[12] D. P. Young, C. M. Keller, D. W. Bliss, K. W. Forsythe, Ultra-wideband (UWB) transmitter location using time difference of arrival (TDOA) techniques, in: The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, vol. 2, pp. 1225–1229.

[13] S. Selvaraj, S. Sundaravaradhan, Challenges and opportunities in IoT healthcare systems: a systematic review, SN Appl. Sci. 2 (1) (2020) 1–8.

[14] Z. A. Khan, P. Herrmann, A trust based distributed intrusion detection mechanism for internet of things, in: 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), 2017, pp. 1169–1176.

[15] A. Froehlich, A. Siebrits, C. Kotze, e-Health: How Evolving Space Technology is Driving Remote Healthcare in Support of SDGs, in: Space Supporting Africa, Springer, 2021, pp. 91–185.

[16] J. Jiang, Z. Li, Y. Tian, N. Al-Nabhan, "A review of techniques and methods for IoT applications in collaborative cloud-fog environment", Security and Communication, Networks 2020 (2020).

[17] M.N. Birje, S.S. Hanji, Internet of things based distributed healthcare systems: a review, J. Data, Inform. Manage. 2 (3) (2020) 149–165.

[18] S. B. Mohanthy et al., Real time internet application with distributed flow environment for medical IoT, in: 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), 2015, pp. 832–837.

[19] E.K. Lee, Y. Wang, R.A. Davis, B.M. Egan, Designing a low-cost adaptable and personalized remote patient monitoring system, IEEE International Conference on Bioinformatics and Biomedicine (BIBM) 2017 (2017) 1040–1046.

[20] S. Roy, A. Rahman, M. Helal, M. S. Kaiser, Z. I. Chowdhury, Low cost rf based online patient monitoring using web and mobile applications, in: 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), 2016, pp. 869–874.

[21] A. M. Ghosh, D. Halder, S. A. Hossain, Remote health monitoring system through IoT, in: 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), 2016, pp. 921–926.

[22] H. Moustafa, E. M. Schooler, G. Shen, S. Kamath, Remote monitoring and medical devices control in eHealth, in: 2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2016, pp. 1–8.

[23] I. A. Pap, S. Oniga, I. Orha, A. Alexan, IoT-based eHealth data acquisition system, in: 2018 IEEE international conference on automation, quality and testing, robotics (AQTR), 2018, pp. 1–5.

[24] M. Hamim, S. Paul, S. I. Hoque, M. N. Rahman, I.-A. Baqee, IoT based remote health monitoring system for patients and elderly people, in 2019 International conference on robotics, electrical and signal processing techniques (ICREST), 2019, pp. 533–538.

[25] L. J. B. Andrews, L. Raja, Remote based patient monitoring system using wearable sensors through online and offline mode for android based mobile platforms, in: 2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions)(ICTUS), 2017, pp. 602–606.

[26] M.W. Iqbal, N. Ahmad, S.K. Shahzad, M.R. Naqvi, I. Feroz, Usability Aspects of Adaptive Mobile Interfaces for Colour-Blind and Vision Deficient Users, Int. J. Comput. Sci. Network Security 18 (10) (2018) 179–189.

[27] Sewisy, Adel, Abd El-Aziz, M. Marghny, Ahmed Taloba, Fast efficient clustering algorithm for balanced data, Available at SSRN 2545138 (2014).

[28] A.I. Taloba, A.A. Sewisy, Y.A. Dawood, Accuracy enhancement scaling factor of Viola-Jones using genetic algorithms, in: In: 2018 14th International Computer EngIneerIng Conference (ICENCO), IEEE, 2018, pp. 209–212.

[29] M. Elloumi, M.A. Ahmad, A.H. Samak, A.M. Al-Sharafi, D. Kihara, A.I. Taloba, Error correction algorithms in non-null aspheric testing next generation sequencing data, Alex. Eng. J. 61 (12) (2022) 9819–9829.