Advances in Computer and Communications, 2023, 4(4), 245-251
https://www.hillpublisher.com/journals/acc/
ISSN Online: 2767-2875

Patient Signs Safety Monitoring System Based on Internet of Things and Cloud Computing

Yiwen Dong*, Leiyufei Wang, Xiaoliang Zhou

James Cook University, Singapore.

How to cite this paper: Yiwen Dong, Leiyufei Wang, Xiaoliang Zhou. (2023) Patient Signs Safety Monitoring System Based on Internet of Things and Cloud Computing. *Advances in Computer and Communication*, 4(4), 245-251.
DOI: 10.26855/acc.2023.08.006

Received: July 8, 2023
Accepted: August 6, 2023
Published: September 5, 2023

***Corresponding author:** Yiwen Dong, James Cook University, Singapore.

Abstract

The contemporary lifestyle has led to a pressing demand for prompt, efficient, and secure medical care. However, the modern healthcare system faces several challenges such as inadequate funding, fragmentation within the medical community, and reduced service efficacy. Innovative solutions, such as the development of an intelligent detection bracelet, have been devised to tackle these issues. The wristband collects health data, encrypts it and transmits it via Internet of Things (IoT) technology to a cloud computing platform for hierarchical management and analysis. Numerous remote health monitoring systems have been developed to facilitate timely notifications and patient feedback, particularly for individuals with deteriorating suboptimal health conditions. The design of these systems aims to ensure proactive measures while minimizing disruptions, thereby enabling individuals to access personalized healthcare support and prompt medical attention through the utilization of IoT technology and remote health monitoring. This effectively addresses resource shortages and enhances overall health outcomes.

Keywords

Remote Health Monitoring, Internet of Things (IoT), Intelligent Detection Bracelet, Healthcare System Challenges, Personalized Healthcare

1. Model or design of the system

In view of the special business scenario of health monitoring system, the following system modes are designed in combination with IOT equipment, cloud platform, encryption calculation, etc., including data such as user's blood pressure, blood oxygen, user's heartbeat quality, etc., which are monitored by IOT equipment, namely the bracelet proposed in this paper, and the above data are divided into emergency data and daily recorded data. Emergency data includes heartbeat quality, and other data are daily recorded data. When emergency data fluctuates violently and abnormally, contact directly through the emergency alarm platform, alarm the user directly, and contact the doctor. Other daily data are encrypted and authenticated by encryption algorithm, and then transferred to the cloud computing platform to form an electronic health record exclusive to the user, and some rights are granted to the doctor. Doctors can check the user's electronic health records and make corresponding guidance and suggestions. The process is illustrated in the subsequent diagram [1].

2. Types of components in system

Among them, the Internet of Things devices use electronic bracelets to record and monitor data. The whole design includes main control chip module, human blood pressure monitoring module, blood oxygen monitoring module, heart rate monitoring module and Lora communication module. The hardware block diagram is shown in the figure.

The auxiliary bracelet software program mainly realizes the functions of real-time measurement of blood oxygen

and blood pressure, real-time heart rate collection, fall detection and alarm, and real-time viewing of server data. The software design block diagram is shown in the figure.

Among them, the motion fall detection device does not give an alarm when the three-axis acceleration sensor is placed vertically, and gives an alarm when the change rate of the sensor's inclination angle reaches a preset value. Through the communication chip, the heart rate, body temperature, data and motion state are transmitted to the wireless gateway, then displayed and stored in the server interrupt [2].

The server judges by receiving data, sends an alarm directly to emergency data, and transmits daily recorded data to the cloud for recording and backup.

Network devices that transmit these include: These are the devices that are equipped with sensors and actuators, and are connected to the network infrastructure through wired or wireless connections.

We also need to consider the life cycle yinsu: The system design must take into account the lifecycle considerations of the IoT devices, including provisioning, patching, and decommissioning of the devices.

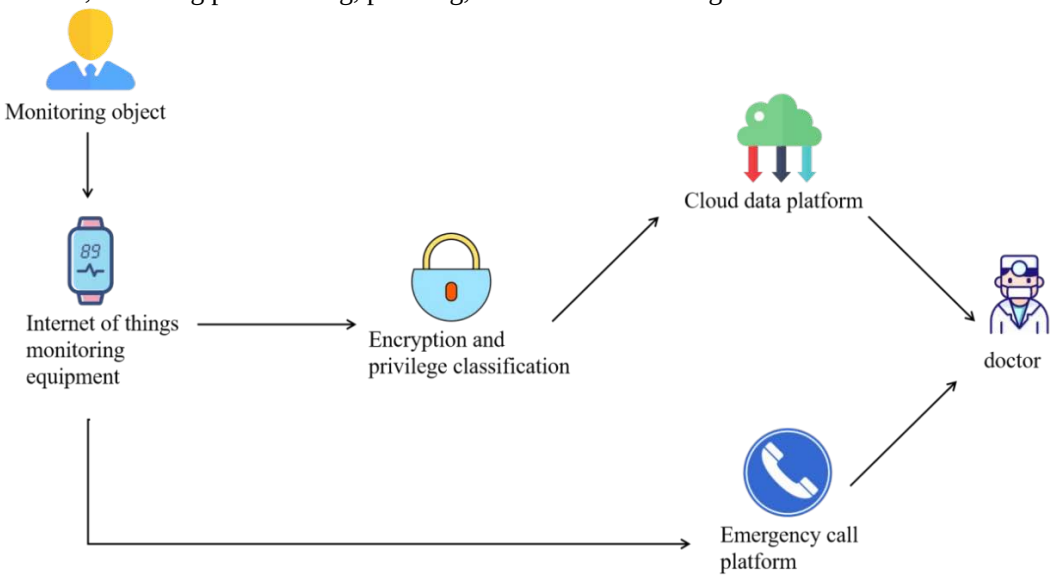

Figure 1. Overall system flow chart.

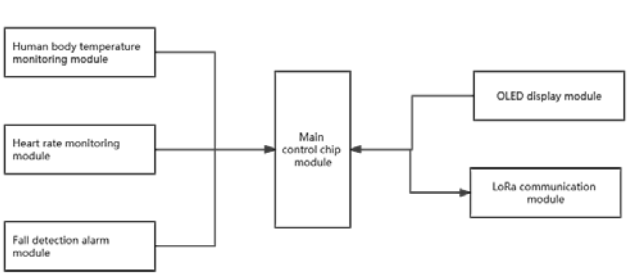

Figure 2. Hardware flow chart.

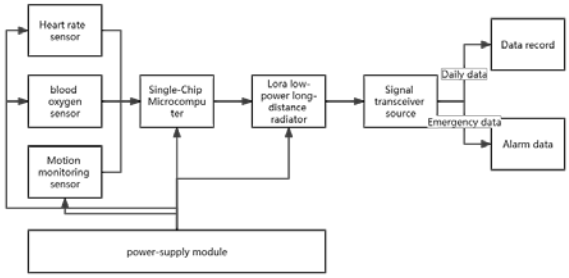

Figure 3. Software flow chart.

3. Processing of cloud platform

When uploading data, first use tcp/udp socket debugging tool, establish a UDP client, select the IP address and set port number of ECS, and conduct data test. After successful communication, run a UDP server program in the cloud, and the platform data flows through the HTTP/2 channel. After configuring the HTTP/2 server subscription, the IOT device will send the message to the server through the HTTP/2 channel. By accessing HTTP2 SDK HTTP/2 SDK to provide authentication, ToDic subscription, message sending and message receiving, MQTT is an instant messaging protocol with low overhead, low bandwidth occupation and real-time and reliable message service for remote devices. Its biggest advantage is that it is widely used in the Internet of Things and embedded mobile devices based on the subscription/publishing mechanism. MQTT can play three different roles: publisher, subscriber and server. MQTT

micro message queue mainly undertakes the work of mobile connection access, connection management and data forwarding, which is equivalent to a connection gateway with unlimited expansion capacity. Sensors send data to the cloud.

The front-end uses Vue.js, JQuery, and Bootstrap frameworks. It separates the front-end and back-end, calling service interfaces provided by the back-end such as user center, information management, authority management, and doctor management to achieve data-driven page response. The back-end service is developed using Google's open source Go language, which handles high concurrency effectively. It adopts a micro-service framework, dividing the system into small basic services that communicate through RPC. This approach reduces module coupling and enhances server stability and flexibility. The main micro-services include: data storage query service, medical service, case service, hospitalization service, settlement service, authority management service and more. The cloud platform provides a fixed IP and port MQTT server as a data forwarding transit station to enable cloud storage functionality. The database in this system manages system information and users' personal daily health records. The saved parameters include blood oxygen, pressure, and heartbeat quality. Abnormal information is detected when the parameters exceed the threshold. System information includes basic user details such as ID, name, phone number, family name and contact number. Account information refers to login credentials.

4. Process to maintain the system to be secure

Considering the variety of user data, different roles only need to view some user data. For example, users can view all their personal information, doctors can only view daily record data, and the alarm center can view emergency alarm data. Therefore, CA certificates are distributed to each role that joins the system for identity authentication. For members with different identities, different user groups and user permissions are set in the cloud processor to meet the hierarchical management of data.

Firstly, CA authentication is distributed. As a trusted third party, the government uses technologies such as public key certificate, digital signature and HSAH function to confirm identity. The specific process is as follows:

A. A role initiates a request to join the system.

B. Government regulatory agencies issue CA certificates. After receiving the application from the server, CA certification agencies will conduct a series of online and offline surveys to evaluate the legality of this applicant. After the CA agencies verify and audit, they will issue CA certificates to the applicants.

The certificate includes the following contents:

1) Version information of the certificate;
2) Serial numbers of certificates, each certificate has a unique serial number;
3) The signature algorithm used by the certificate;
4) The name of the issuer of the certificate, and the naming rules generally adopt X.500 format;
5) the validity period of the certificate, the universal certificate generally adopts UTC time format;
6) the name of the certificate owner, naming rules generally adopt X.500 format;
7) The public key of the certificate owner;
8) Signature of the certificate issuer.

In order to prevent certificate forgery, digital signature technology is introduced here. The so-called "digital signature" is to generate a series of symbols and codes through some cryptographic operation to form an electronic password for signature instead of writing a signature or seal. The algorithm process is as follows:

1. Send the message: the sender uses a hash function to generate a message digest from the message text, and then encrypts the digest with his own private key. The encrypted digest will be sent to the receiver as the digital signature of the message and the message [3].

2. Receiving message: The receiver first calculates the message digest from the received original message with the same hash function as the sender, and then decrypts the digital signature attached to the message with the sender's public key [4].

Here, DSA algorithm is specifically used for signature:

Main parameters of DSA:

Global public key component, which can be shared by users.

P: prime number, which requires 2L-1<p<2L, 512<=L<1024, and L is a multiple of 64.

Q: the prime factor of (p-1), 2159<q<2160, that is, the bit length is 160 bits. G: = h (p-1)/qmod p. where h is an integer, 1<h<(p-1) and üh(p-1)/q mod p>1. User private key

X: random or pseudo-random integer, requiring 0<x<q User public key
Y： =gx mod p
Random number k
Random or pseudo-random integer, requiring 0<k<q Then company A randomly selects K.
Calculate e = h (m);
Calculate r=(gk mod p) mod q Calculate s = k-1 (e+x r) mod q
Output (r, s) is the digital signature of message m.

After receiving M, R and S, the receiver B first verifies that 0 < R < Q, 0 < S < Q. Calculate e = h (m);
Calculate w=(s)-1 mod q Calculate u1 = e w mod q Calculate U2 = r w mod q
Calculate ① v = [(gu1 yu2) mod p] mod q.
If v=r, the signature is confirmed to be correct, otherwise it is rejected.

If the digests match, the receiver can verify the digital signature belongs to the sender. By comparing abstracts of A and B, we can determine if the content is complete. Decrypting with A's public key confirms it was sent by A (private key encryption). The information digest here refers to the extraction of fingerprint information (digest information) from all data using a data digest algorithm, which enables functions such as data signature and integrity check. Due to its irreversibility, it is sometimes used for encrypting sensitive information. In order to judge the information digest, the original text is converted into a unique value with a fixed length by using a hash function. The same text hashed with the same algorithm will always generate the same hash value, which can be used as a unique identifier for its associated data. Specifically, SHA-2 algorithm is utilized here:

SHA-256, which processes the packet size of 512 bits and generates a message digest of 256 bits.
SHA-224, which processes the packet size of 512 bits and generates a message digest of 224 bits.
SHA-512, which processes the packet size of 1024 bits and generates a message digest of 512 bits.
SHA-384, which processes the packet size of 1024 bits and generates a message digest of 384 bits. In this way, the CA generates a certificate with a unique digital signature, and after the data is stored in the cloud platform, hierarchical authority management is required. When setting user groups, the following methods are adopted:

Add one or more users to a user group, which is uniquely identified by GID (group identifier). Set the administrator to combine the general group, which is divided into user group, doctor group and alarm platform group.

- Administrator group: root-0
- General group:
- System group: allocate rights to resources obtained by daemons.
- Ordinary group: for users.

Each user is uniquely identified by UID (user identifier). The permissions of the file for visitors are divided into:

- R: readable
- W: writable
- X: executable □

According to different user rights, the displayed data is different.

The data transmission process employs asymmetric and symmetric encryption to ensure secure transit and storage, while sensitive data is further protected with MD5 hashing for enhanced security in the smart medical system. The encryption process involves two parts: asymmetric encryption with public key X for user data, followed by symmetric encryption with key B. Upon receipt by the cloud platform, private key is used to decrypt key B and then decrypt the data using key B.

The asymmetric encryption algorithm, known as key encryption, uses two keys: a public key and a private key. Content encrypted with the public key can only be decrypted with the private key, while content encrypted with the private key can only be decrypted with the public key. Therefore, the RSA encryption algorithm is specifically chosen. The first task of RSA is to choose two big prime numbers p, q;

Calculate n = p*q, and n is the key length. The larger this value, the safer it is.

Calculate the Euler function of n (n) = (p-1) (q-1);
Choose e, e satisfies 1 < e < (n), and gcd [e, (n)] = 1 (that is, e and (n) are prime numbers);

Find d according to the formula e * d ≡ 1mod [(n)]

Encapsulate n and e as public keys and n and d as private keys. The public key is published to all users who want to send messages to them, and the private key is kept secret.

The data between companies A and B is transmitted in plaintext, using a symmetric encryption algorithm for data encryption. A key has been designed by both companies to encrypt the transmitted data. With symmetric encryption, information can be encrypted or decrypted. In this case, the RC4 algorithm is chosen for data encryption, following the process below.

Initialize the S table: linearly fill the S table (256 bytes), circularly fill another 256- byte K table with the seed key, and initially replace the S table with the K table.

Generation of key stream: a pseudo-random number is generated for each byte to be encrypted, which is used for XOR. Once the initialization of table S is completed, the seed key will no longer be used.

The decryption process is XOR twice, so you can get the original as long as you take the key stream back and XOR it once.

5. Process to maintain the system to be secure

Threat modeling of IOT devices;

Firstly, assets that may need to be protected include:

- firmware
- Certificate and device unique key
- Login credentials (user or administrator)

System configuration (to ensure that your IP will not be damaged or control will be taken away)

- Event log
- Health data

The list of assets listed in network communication may not be exhaustive, but it will include the most valuable assets or data for users. Then analyze potential opponents, attackers and threats:

In the health system of, attackers are divided into the following categories:

- Remote software attacker
- Network attacker
- Malicious internal attacker
- Advanced hardware attacker The threats include:
- Deceive identity
- Tampering with data
- deny
- information disclosure
- Denial of service
- privilege escalation

If the user equipment is taken as the entry point, the potential communication attacks may include:

- Deception, that is, an unauthorized person who impersonates a legitimate user to access equipment.
- Permission is elevated, or an attacker tries to destroy voice ID authentication to identify it as a legitimate user to destroy the system.

In the case of a network connected to a cloud server, threats that may be considered include:

- Deception, illegal access to equipment to use the victim's authentication information
- Tampering with data
- Information disclosure means releasing information that should be kept confidential, such as user credentials.
- Denied service to valid users. This may threaten availability and reliability or temporarily disable the device.
- Elevated privileges or an attacker who tries to log in as an administrator to gain access or control of the

device.

High-level security objectives to deal with threats by setting security objectives. Taking IOT devices and cloud platforms as examples, high-level security objectives can include:

- Security identity
- Secure Boot and Firmware Upgrade
- Security audit
- Secure storage and binding
- Defence in depth
- Safety life cycle management

To sum up the above, all the information collected can now be merged into the threat summary table, and a separate summary table can be created for each asset identified earlier, as shown in the following figure:

Table 1. Threat summary table

Asset	Threat	Impact	CVSS	Counter-measure on Threat	Arm's Solution
Firmware	Tampering	Execute malicious code Bypass secure boot Extract sensitive information Physically install malware	Critical: 9.0 High: 7.1	Support secure boot flows and firmware authenticate Prevent voltage/current glitches and debug bypass	[CC]Root of Trust [CC]Loaded SW validation [PSA]Trusted Boot features [CC]SW update validation [CC]Rollback pratection [PSA]Firmware Update features [SDC]Secure debug channel
	Escalation of privilege	Launch DDoS attack Tamper/Steal voice records	Critical: 9.0	Enforce principle of least privilege	[Tz], [CI] Secure partitions & Isolation
	Information Disclosure	Explore firmware vulnerability	Medium: 4.8	NSPE isolation Execute-only RAM Firmware encryption	[CC]Secure crystallographic and RNG support
	Dental of service	Permanent bricking of system Cause the device to catch fire - thermal control Drain battery	Critical: 9.0	Disaster recovery Enforce access control to critical resources	[CC]Secure crystallographic and RNG support [TZ]TA to manage sensitive Operations [TZF]Use trust zone filters to enforce access controls

6. Threat modeling of the system

With the rapid development of Internet of Things, cloud computing and other technologies, health data monitoring becomes easy. Based on Internet of Things, cloud computing, network security and other technologies, this paper proposes to access people's health and activity information, location, environment and other surrounding related information into the network system for real-time monitoring, data storage and processing, and information service. Through efficient system integration and management, it reflects the caring service for the ward and provides users (including ward, guardian and staff) with more efficient, convenient and safe services. However, in the cloud platform data computing, data encryption and other parts, we can continue to study in depth [5].

References

[1] Mamdiwar, S. D., R, A., Shakruwala, Z., Chadha, U., Srinivasan, K., & Chang, C. Y. (2021, October 4). Recent Advances on

IoT-Assisted Wearable Sensor Systems for Healthcare Monitoring. MDPI. https://doi.org/10.3390/bios11100372.

[2] H., Liu, N., Zhang, D., Su, Z., & Wang, T. (2021, February 26). Preimpact Fall Detection for Elderly Based on Fractional Domain. Preimpact Fall Detection for Elderly Based on Fractional Domain. https://doi.org/10.1155/2021/6661034.

[3] Dasgupta D, Shrein JM, Gupta KD. (2019). A survey of blockchain from security perspective. Journal of Banking and Financial Technology, 3(1):1-17.

[4] Digital Signature Algorithm (DSA) in Cryptography: A Complete Guide | Simplilearn. (2021, July 29). Simplilearn.com. https://www.simplilearn.com/tutorials/cryptography-tutorial/digital-signature-algorithm.

[5] Butpheng, C., Yeh, K. H., & Xiong, H. (2020, July 17). Security and Privacy in IoT-Cloud-Based e-Health Systems—A Comprehensive Review. MDPI. https://doi.org/10.3390/sym12071191.