Advances in Computer Science Research, volume 80
International Conference on Computer Science, Electronics and Communication Engineering (CSECE 2018)

Networking Mechanism of Fire Monitoring System Based on Internet of Things

Fangwen Xu[1], Jianjun Yi[1,*], Liang He[2,3], Xiaomin Zhu[1] and Zhuoran Wang[1]
[1]Department of Mechanical Engineering, East China University of Science and Technology, Shanghai 200237, China;
[2]Shanghai Aerospace Control Technology Institute, Shanghai 201109, P. R. China
[3]Shanghai Key Laboratory of Aerospace Intelligent Control Technology, Shanghai 201109, P. R. China
[*]Corresponding author

***Abstract*—Nowadays, there are some problems exist in remote monitoring of IoT fire control system, such as bad real-time performance, difficulty access of massive nodes, the security problems arisen due to the lack of standardized authentication and registration mechanism. In view of the problems above, this paper proposes a networking mechanism of fire monitoring system based on IoT. This mechanism uses public IoT cloud hosting service to meet the needs of mass nodes access. System architecture is Publish/subscribe mode based on lightweight MQTT protocol, so it can meet the real-time data transmission. We add the node certification mechanisms in the process of node registration and encrypt the message in the process of transmission. This standardized network-access mechanism improves the security of the whole system. Through short-term testing, this network mechanism can well solve the above problems in the current fire monitoring system. Due to its scalability and high stability, this network mechanism can be widely used in online monitoring of large-scale public places and buildings.**

Keywords—Network Mechanism; MQTT; registration; authentication

I. INTRODUCTION

With the rapid economic development in our country and the continuous expansion of urban construction, the number of high-rise, super-tall buildings and large-scale buildings in cities has been increasing. However, due to the high floors, complicated functions, numerous facilities and staff, the hidden danger of causing fires also increases greatly. Therefore, fire prevention work which relates to the safety of people's life and property becomes increasingly important. It is necessary to develop the fire Internet of Things technology platform[1].

This paper focuses on networking mechanism based on the Internet of Things fire monitoring system. It can meet the flexible access of massive nodes and give access to alarm of fire facilities, flow and pressure parameters, temperature and equipment voltage parameters. We collect, process, analyze the real-time useful information during the reliable transmission and realize the effective perception of fire safety management information. [2].

Nowadays, the fire monitoring system based on IoT in the market is still in infancy.Most of the system applies the request/reply mode, which is synchronous. The publish/subscribe mode with the advanced IoT protocol is rarely used. However, this mode decouples the relationship between the customer (publisher) who post the message and the customer (subscriber) who subscribe to the message. In addition, most of the fire monitoring systems at this stage merely realize the data transmission and seldom consider the cyber security of the system. They do not have a standard registration and authentication mechanism, which makes the node management of the system nonstandard and it is the very easy to eavesdrop the transmitted data. And the system can be also easily broken paralyzed by hackers' disguised nodes through the information bomb and denial of service means. So it is particularly important to consider of the system safety and standardized management in firefighting field, which is an important public utility involving the life and property of the masses. Therefore, this paper presents a networking mechanism of fire monitoring system based on the Internet of Things from the perspective of cyber security, management standardization, remote monitoring and maintenance of firefighting facilities[3].

This article is divided into four chapters. The second chapter describes the Internet of Things fire monitoring information system overall framework. The third chapter introduces the main technical realization of IOT fire monitoring information system in detail, including the IoT protocol selection, the process of node authentication registration and the design and implementation of the edge node. The fourth chapter summarizes this article.

II. THE OVERALL FRAMEWORK OF THE FIRE MONITORING SYSTEM BASED ON THE INTERNET OF THINGS

The architecture of the fire monitoring system based on the Internet of Things includes the perception layer, transport layer, storage layer and application layer.

The perception layer is mainly used for information collection for various types of fire equipment and environmental parameters. The data collected will be transmitted according to pre-defined protocol including the node number and time stamping. At the same time, combined with RFID technology, fire extinguishers, fire hydrants and otherfirefighting facilities can be located and position information of them will be real-time transmitted in order to achieve the intelligent monitoring and maintenance of the fire service facilities.

The transport layer applies the SSL protocol in the way of digital certificate and key encrypt to complete node and cloud two-side authentication and data encryption. The authenticated node will successfully register into the fire monitoring system,

and encrypt and package the data collected from the sensing layer, and publish it to the cloud through WIFI or Ethernet network via the MQTT protocol so that the cloud information can be collected and shared. Considering that there are some environments without WiFi or Ethernet networks, we construct the WSN network using Zigbee's ad hoc networking technology in this scenario. The information from the sensing layer is transmitted to the gateway via the MQTT-SN application layer protocol which is transplanted to the Zigbee protocol stack. In the gateway side, we achieve the protocol conversion between MQTT-SN and MQTT, and then transmit the data to the cloud via MQTT[4].

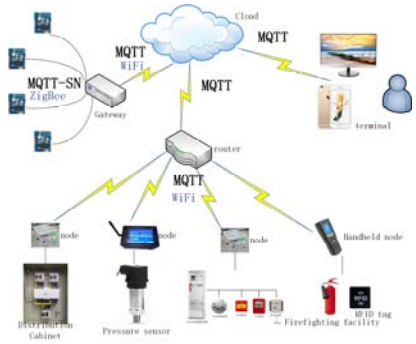

FIGURE I. THE ARCHITECTURE OF THE SYSTEM

The storage layer uses cloud hosting services so that we just focus on the IoT business.The system will realize the cloud-based MQTT hosting services by using the public cloud IoT HUB service. And the application layer mainly realizes the interaction between the monitoring system and the user. The management staff or other staff members use the MQTT protocol in the terminal to obtain the real time or the historical information of the fire system.

III. THE MAIN TECHNICAL IMPLEMENTATION OF THE NETWORKING MECHANISM

The innovation of this system lies in the use of lightweight MQTT protocol, which supports publish/subscribe message push mode and ensures the real-time transmission of system data. At the same time, The node registration and authentication mechanism is introduced in this mechanism in order to ensure the security and reliability of the system. We will separately introduce the use of the MQTT protocol, node certification registration mechanism and the design and implementation of the Internet of Things edge below.

A. MQTT-based Publish/Subscribe Message Push Mode

MQTT (Message Queuing Telemetry Transport) is an instant messaging protocol developed by IBM that promises to be an important part of the Internet of Things. It can complete instant messaging in case of less occupied resources than other protocols[5].Compared with HTTP, CoAP, XMPP and other protocols, MQTT protocol has the following advantages:

- MQTT is based on TCP and is more reliable
- MQTT realizes Publish / Subscribe mode in an asynchronous way.
- MQTT provides many thoughtful designs for Internet of Things, such as Quality of service and "last will."
- MQTT is a binary format which is lighter than XMPP.

Instead of requesting/answering such a synchronization mode, the publish/subscribe model decouples the relationship between the client (publisher) who posts the message and the client (subscriber) who subscribes to the message. So there is no need for publisher and subscriber to establish contact directly[6].

MQTT categorize messages by a UTF-8 string topic, which can represent multiple levels of relationships through backslashes. Topics can be filtered by wildcards. Among them, '+'can filter one level, while '#'can only appear at the end of the topic to filter any level [5].According to the fire monitoring situation, we customize some topics like below:

- building-b / floor-5: Representing the devices on the fifth floor in Building B.
- + / floor-5: Representing the devices on the fifth floor in any building.
- building-b / #: Representing all the devices in building B.

B. The Edge Node's Authentication Registration Process

The heart of the IoT is to make life more comfortable and convenient by exchanging and analyzing data from connected everything. However, unpredictable adverse consequences result from the disclosure of sensitive data and the unlawful control of devices. For the utility like fire protection, which is related to people's life and property safety, the consideration of IoT security is particularly important. The cyber security in this system is ensured by two ways. One is to realize the data encryption communication between the nodes and the cloud. The other is the authentication of edge node's identity and the cloud's identity. The following is the flow of device registration process to the cloud.

The system in our paper takes advantage of the MQTT security features at the transport and application layers. Using SSL, a very sophisticated security protocol to create a secure connection when nodes are handshaking, so that it can prevent hackers from eavesdropping or tampering with the messages. Ensuring the identity of each node (authentication) will be an extremely important consideration for IoT network security.The security of the IoT edge nodes includes confidentiality and integrity. The common way is the use of public keys or private keys as a unique part of the authentication and identification process.An authenticated node helps the system designers ensure that these devices meet all of the required standards without risking the entire network ecosystem.

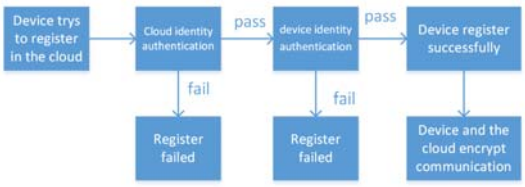

FIGURE II. THE NODE'S REGISTRATION PROCESS

Cyber security is the foundation for a successful deployment of the Internet of Things. At present, the edge node is the weakest part of the network to ensure IoT security, while the protection of encryption key can help ensure the edge node identity. The best way to achieve this is to use the protected hardware. According to the research on the reliability and the cost of the security IC in the market, our edge node uses Atmel ATECC508 to implement encryption function. Atmel ATECC508A is the first IC to integrate the ECDH (Elliptic Curve Diffie-Hellman) security protocol and provides reliable security for the Internet of Things (IoT) market such as home automation, industrial networking, accessories and supplies verification. ATECC508 integrated with ECDH and ECDSA which can effectively provide superior confidentiality, data integrity and authentication capabilities for the MCU or MPU which runs encryption/decryption algorithms such as AES in the software. [7].

Each ATECC508 chip has a pair of unique public and private keys in the factory time. And each edge node in the system is assigned with a unique UUID. Therefore, during the production of the edge node, the UUID, the public key and private key of the node are one-to-one and unique. Based on this feature, the UUID is used as the starting point to implement the process of the register and authentication of the edge node later[8].

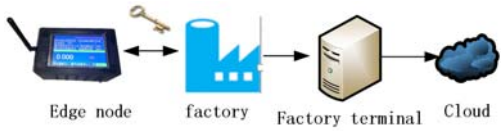

FIGURE III. THE PUBLIC KEYS STORAGE FLOW IN THE FACTORY TIME

FIGURE IV. THE AUTHENTICATION PROCESS OF THE CLOUD IDENTITY

The workflow of the edge node in the production time is shown in the figure. After the edge node has been uploaded the factory time code, the ATECC508 chip of the node will generate a pair of public key and private key. The MCU reads the public key of the ATECC508 chip via the IIC interface and sends the public key and the corresponding UUID of the edge node to the receiving terminal of the factory. The receiving terminal is used to process the public key and the UUID information of all nodes from the factory and then pack all of the information and upload them to the cloud platform database for storage. These data will be used in the later authentication.

When the nodes finish the factory process, they should be re-burned the application stage code before they are put into formal use. In the application process of the edge node, the authentication and registration process of the node mainly includes the cloud identity authentication and the node identity authentication.

When the edge node connects to the network and tries to link the cloud for the first time, the cloud will send its own certificate information through the SSL protocol. After the node receives it, it will decrypt the cloud certificate according

to the CA public key storage in the firmware of the node in order to get the cloud Public key. The node then will send a random string to the cloud to encrypt using the cloud private key. The cloud will return the encrypted message to the node.

And the cloud public key decrypted before will be used by the node to decrypt the returned message. If the decrypted message is the same as the random string generated before, the owner of the private key is really the cloud that certifies the certificate. After the authentication confirms the identity of the cloud, the identity of the node which is attempting to register will be certified next.

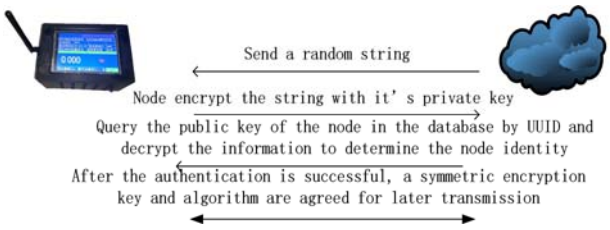

FIGURE V. THE AUTHENTICATION PROCESS OF THE NODE IDENTITY

The cloud sends a random string to the node. The node encrypts it with its own private key and returns it to the cloud. The cloud uses the node's UUID to query the public key stored in the cloud database and use this public key to decrypt the message. If the decrypted string is the same as the previous one sent before. The identity of the node is confirmed and the handshake process finishes successfully, allowing the edge node's registration in the cloud. After the registration process finishes successfully, the node will generate a random symmetric key. Then the node will encrypt the symmetric key and its corresponding algorithm using its own private key and sends the encrypted messages to the cloud. The cloud decrypts the encrypted messages using the node's public key and gets the symmetric encryption key and algorithm agreed by both parties. So, only the node and the cloud these two parts know the agreed symmetric key and algorithm. In this way, the messages encrypted by this symmetric key and algorithm will be protected from being eavesdropped and lost.

C. The Design and Implementation of the Edge Node

In order to adapt to different environments, we may use WIFI or Ethernet these two ways to communicate. Therefore, in the design of the edge node, we should consider both WIFI and Ethernet communication module integration.

The main chip is the stm32F4 chip of Cortex M4 core. We use the ESP32WIFI module as the wireless communication module and realize Ethernet transmission by using the W5500 chip to complete SPI to Ethernet protocol conversion. The main chip connects the encryption chip ATECC508A by IIC to realize the generation of key pairs and private key storage. An external flash chip is used to store the CA public key information used to

check the cloud certificate, the cloud connecting URL address, and the configuration information of the WiFi and Ethernet modules, and so on. The physical appearance of the external node is shown below.

FIGURE VI. THE PHYSICAL APPEARANCE OF THE NODE

FIGURE VII. THE NODE'S LOG FROM THE UART

The implementation of the nodes' MQTT transplantation in software is mainly in two aspects: One is ESP8266 IDE 2.0 MQTT protocol development for ESP8266 module and the other is the development of the application layer MQTT protocol based on Ethernet transmitting in stm32 IDE.

I use the public MQTT broker CloudMQTT to test the network access function of this node. The configuration information like router name, password, broker URL, port and user name and password registered in the broker are set in the program. After uploading the compiled file into the module, the node successfully run as shown. We can see the Log information from the output of the node serial port. The node successful connect to the MQTT cloud broker. and the node and the node successfully subscribed and released three topics. And the same time, the cloud could receive the message posted by the node in real time.

IV. SUMMARY

In this paper, we propose a networking mechanism of fire monitoring system based on IoT according to the problems exist in remote monitoring of IoT fire control system. The hardware and software development of the edge node have been implemented. The feasibility of this networking mechanism is verified through the test between the edge node and MQTT broker. Because of the scalability, high stability of the network mechanism, it can be widely used in large-scale public places and buildings online monitoring.

Due to the time constraints, there is still room for improvement. For the scenario that does not cover the WiFi network, this article just proposes a feasible scheme framework but does not realize and demonstrate some specific details in the scheme. In the follow-up research, I will conduct an in-depth research on network mechanism of-of massive heterogeneous nodes.

ACKNOWLEDGMENTS

This paper was supported by the Research Foundation of Science and Technology Commission of Shanghai under Grant No. 10DZ1500200, the Natural Science Fund of China (NSFC) under Grant Nos. 51575186, 51275173, and 50975088, the Fundamental Research Funds for the Central Universities under Grant No. WH0913009, Shanghai Pujiang Program under Grant No. PJ201000353, and Shanghai Software and IC industry Development Special Fund under Grant No. 120493.

REFERENCES

[1]. A. Foster, "Messaging technologies for the industrial internet and the internet of things whitepaper," PrismTech, 2015, 23 (1): 5-8

[2]. B.-C. Chifor, I. Bica, V.-V. Patriciu, F. Pop, A security authorization scheme for smart home Internet, 2015, 57 (1): 5-8

[3]. Internet Of things devices, Future Gener. [J]Computer.Syst. 2017, 57 (2): 7-10

[4]. Daniel Barata, Andreia Carreirob. System of acquisition, transmission, storage and visualization of Pulse Oximeter and ECG data using Android and MQTT [J]. Procedia Technology 9 (2013) 1265–1272

[5]. Komkrit Chooruang, Pongpat Mangkalakeeree.Wireless Heart Rate Monitoring System using MQTT [J]. Procedia Technology 86 (2016) 160 -163

[6]. Rui-xiang Chen,Xiao-qiang Zhang,Chao-yang Peng,Hong-yong Zhang(2013). A Study on Design and Implementation of Remote Fire Monitoring System for Buildings[J].Procedia Engineering, 52,56 – 59.

[7]. Ke Yin, JunCheng Jiang(2014). an Application of Internet of Things in the Field of Urban Building Fire Safety[C].International Journal of Safety and Security Engineering,4(2),135-142.

[8]. B. B Prahlada Rao, Cloud computing for Internet of Things & sensing based applications ,Sensing Technology (ICST), 2012 Sixth International Conference on 18-21 Dec. 2012, Kolk