Regular paper

Journal of the KIECS. pp. 101-108, vol. 16, no. 1, Feb. 28. 2021, t. 105, pISSN 1975-8170 | eISSN 2288-2189
http://dx.doi.org/10.13067/JKIECS.2021.16.1.101

# OneNet 클라우드 컴퓨팅 기반 실시간 홈 보안 시스템

김강철* · Yongjiang Zhao**

## OneNet Cloud Computing Based Real-time Home Security System

Kang-Chul Kim* · Yongjiang Zhao**


## 요 약

본 논문은 스마트폰으로 집 내의 상태를 제어하는 OneNet 클라우드 플랫폼 기반 실시간 홈 보안 시스템을 설계한다. 제안된 시스템은 로컬과 클라우드 지역으로 구분된다. 로컬 지역은 I/O 디바이스, 라우터와 센서 데이터를 수집, 모니터링하고 클라우드로 데이터를 전송하는 라즈베리파이로 구성되며, 라즈베리파이에 플래스크 웹 서버가 구현된다. 사용자가 집에 있을 경우 플래스크 웹 서버를 통하여 직접 데이터에 접근할 수 있다. 클라우드 지역에서 사용되는 플랫폼은 중국 통신회사의 OnetNet이며, 원격 접속 서비스를 제공한다. 스마트폰에서 사용자와 홈보안 시스템 사이의 통신을 위하여 하이브리드 앱이 개발되고, 센서 데이터와 비디오스트림을 전송하기 위하여 EDP와 RTSP 프로토콜을 파이썬 언어로 구현한다. 구현된 시스템에서 사용자는 스마트폰으로 센서 데이터. 비디오스트림과 위험이 발생시에 경고 문자를 받을 수 있고, OneNet 클라우드를 통하여 원격으로 집 내의 상태를 모니터링하고, 제어할 수 있다.

## ABSTRACT

This paper builds a real-time home security system based on the OneNet cloud platform to control the status of the house through a smartphone. The system consists of a local part and a cloud part. The local part has I/O devices, router and Raspberry Pi (RPi) that collects and monitors sensor data and sends the data to the cloud, and the Flask web server is implemented on a Rasberry Pi. When a user is at home, the user can access the Flask web server to obtain the data directly. The cloud part is OneNet in China Mobile, which provides remote access service. The hybrid App is designed to provide the interaction between users and the home security system in the smartphone, and the EDP and RTSP protocol is implemented to transmit data and video stream. Experimental results show that users can receive sensor data and warning text message through the smartphone and monitor, and control home status through OneNet cloud.



## 키워드

Home Security System, Raspberry Pi, Flask Web Server, OneNet, EDP,, RTSP protocol
홈보안시스템, 라즈베리파이, 플래스크 웨서버, OnetNet, EDP, RSTP 프로토콜


* 교신저자 : 전남대학교 전기전자통신컴퓨터공학부
** 전남대학교 대학원 컴퓨터공학과 석사과정 (2481985743@qq.com)
· 접 수 일 : 2020. 11. 10
· 수정완료일 : 2020. 12. 29
· 게재확정일 : 2021. 02. 17

· Received : Nov. 10. 2020, Revised : Dec. 29, 2020, Accepted : Feb. 17, 2021
· Corresponding Author : Kang-Chul Kim
School of Electricity, Electronic communication and Computer Engineering, Chonnam National University,
Email : kkc@jnu.ac.kr

## Ⅰ. Introduction

The IoT(Internet of Things)[1] is the most popular technology globally in the 4th industrial revolution society and changes people's lifestyles more convenient. IoT refers to network identification and management through the massive combination of devices and the Internet. With the continuous development of IoT technology and the growth of the market scale, it has become one of the core technologies in the home security system.

With the rapid development of the economy and the increased urban population dramatically, theft, burglary, robbery, and other incidents rose. These incidents take severe influences on the stable life of people. People want to check home situations at any time in any place. As the technology of communication, sensors, and IoT has developed, home security has entered the era of intelligence. An intelligent home security system collects data from sensors and sends the information to the user via the Internet, no matter where the user is. The user can monitor the situation of a home with a smartphone[2].

The home security system, including sensors and actuators, progresses rapidly, but the volume of data they generate is challenging to store and process on local platforms. Cloud computing's scalability provides a solution to this problem and resources at a low cost for its users. OneNet is an open-source cloud platform of China Mobile. The services are free, easy to use, and has high practicality and value for learning and research.

This paper proposes a home security system based on OneNet[3]. The system is mainly composed of OneNet, Flask[4], and RPi(Raspberry Pi). When the user stays outside of the home, OneNet provides a remote connection between the user and the RPi through the EDP[5](Enhanced Device Protocol) protocol. RPi connects to OneNet through a Wi-Fi router, collects sensor data, and sends the data to OneNet. Flask is a local server running in RPi. A user accesses Flask directly to obtain data and control the device at home. To enable the user to see the situation at home, RTSP[6](Real-Time Streaming Protocol) protocol is implemented for video transmission. The user can watch a video through the smartphone. When an abnormal event happens at home, the home security system sends an alarm message to the user.

This paper is organized as follows: Section Ⅱ presents the related work. In section Ⅲ, the system overview is described. In Section Ⅳ, system implementation is described, and the system results are shown in section Ⅴ. Finally, the conclusion is described in section Ⅵ.

## Ⅱ. Related works

The smart home security system has developed rapidly in recent years, and many solutions have been proposed.

R.Piyare proposed a BT(Blue Tooth) based home automation system that uses Arduino as the controller and the relay as the actuator to build a simple control system[7]. A user can control home devices with a smartphone, but the paper has a disadvantage that user repeats the operation without knowing the state of the device because the system cannot obtain the running status of the appliance before sending the control command.

R.Teymourzadeh[8] proposed Smart GSM(Global System for Mobile communications) based Home Automation System. The system uses PIC16F887 as a controller, connects the GSM module through RS232, and an actuator is a relay. The user can send SMS(Short Messaging Service) to the GSM module through the smartphone. SMS module carries control commands to check the running status of the device. GSM signals are not very good when a house has many rooms, and GSM incurs some fees.

D.Sunehra[9] proposed an interactive home automation systems based on email and Bluetooth technologies. It uses RPi as the gateway, Arduino as the controller, and Arduino is connected with a relay to control home devices. RPi connects to the router via wireless. User can send emails to RPi to control and monitor the running status of home devices when the user stays at home, and the user can control home devices efficiently through BT. But controlling the device by sending the email is not very convenient for the user.

J.Du[10] proposed a remote monitoring system of temperature and humidity based on the OneNet cloud service platform. The system uses a cloud platform and gives an excellent solution for users′ demand remotely controlling home devices. Users can control the equipment′s status and monitor the data in real time through the cloud platform anywhere. But when the temperature or humidity is abnormal, the user cannot watch the home through the camera.

S.Jindarat[11] proposed a smart farm monitoring using RPi and Arduino. When an exception occurs, the camera records the environment by taking photos and sending photos to the user. But the camera only takes pictures, and no video is provided. The user cannot see the video of the home in real time.

Users want to get the data from sensors and watch the video of the house in real time and save the data and video in a cloud to use them anytime and anywhere. And it is more convenient to monitor and control the house in a web App through a smartphone.

## Ⅲ. Proposed System Overview

The proposed IoT system has four layers of deployment, devices, gateway, cloud, and application layer as shown in Fig.1.

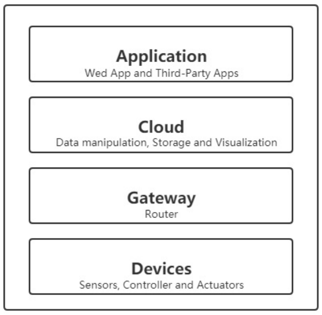


그림 1. 제안하는 시스템 계층 구성
Fig. 1 Proposed system deployment

The device layer includes sensors, controllers, and actuators. The sensor collects data and communicates with the controller. The controller is used to control home devices with actuators, save the data in a local database, and communicate to a router. The controller has a web server and a local database.

The gateway layer has a router for the exchange of data over the Internet. The data is sent to the cloud through the router,

The cloud layer includes data manipulation, storage, and visualization. The cloud provides API(Application Programming Interface) for the devices layer and application layer access. The devices and App can transfer data through the EDP protocol after successfully connecting through the API. Cloud provides a large amount of data storage space and management equipment. Large amounts of log data at the device layer can be stored in the cloud, and the user can also control devices through the cloud.

The application layer includes web App and third-party App. The web App allows the user to visualize the data of the device and control it at the same time. Third-party App is visual interfaces provided by cloud platforms to manage devices.

The user can use Android or iOS phones with the cloud's API to get information from the cloud through web App and control devices by pressing buttons in the hybrid App.

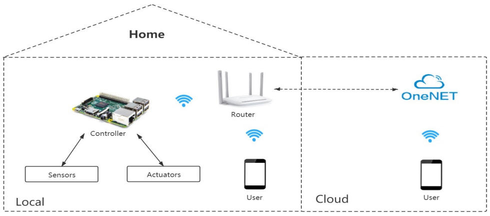


그림 2. 제안된 시스템 구조
Fig. 2 Proposed system structure

The proposed system structure is divided into the local part and the cloud part as shown in Fig.2. The local part has a controller, sensors, actuators, and router in the environment in which the user stays. The controller is RPi that collects the data of sensors and controls actuators. The router is used for the network connection between RPi and the user. When the user sends an HTTP(Hyper Text Transfer Protocol) request to get the home data from the RPi with the flask server is running, the RPi responds the data to the user through an HTTP response, and then the user can view the home data and control the actuators through the smartphone in real time. The RPi collects data from sensors and saves in the SQLite database for the convenience of querying historical data.

The cloud part is mainly the cloud platform of OneNet. The data is uploaded to the OneNet via the EDP protocol from RPi in a local part. The user obtains the data of the sensors and sends the command to control the actuators. OneNet cloud computing, in which several servers and processes are converged on one cloud platform, allows users and RPi to share resources and can be accessed at any time and any place. The cloud environment can be used to store or analyze the transmitted data. The control panel is implemented as a web interface and is equipped with some visualization ability. This enables visualization of historic data readings and analyzed information.

## Ⅲ. System Implementation

### 3.1 Flow chart for cloud system

OneNet platform provides the API based on the HTTP and RESTful[12] by EDP protocol used for device access, encrypted transmission and data storage, and other functions. RPi establishes TCP(Transmission Control Protocol) network connection with the cloud platform through a Wi-Fi module with the router and transmits data to the cloud platform through the EDP protocol as shown in Fig.3.


그림 3. OneNet 통신 절차
Fig. 3 Communication diagram with OneNet

EDP protocol developed by the OneNET platform realizes data forwarding of two EDP devices, that is, point-to-point communication. The message consists of three parts: a mandatory message header, multiple optional options, and an optional message body. The workflow of EDP is shown in Fig.4.

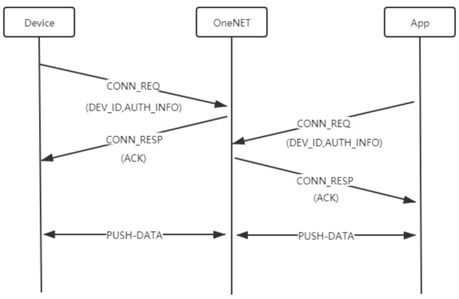


그림 4. EDP 프로토콜 흐름도
Fig. 4 EDP protocol flow diagram

```
1 RPi
HOST ← "jjfaedp.hedevice.com"
PORT ← 876
DEV − IDT ← "559451901"
API − KEY ← "***************"
Socket ← connectOneNet()
receivedData ← Socket.receiveFromOneNet()
Socket → edpConnectOneNet(HOST, PORT, DEV − ID, API − KEY)
```

```
2 Thread1: RPi sends data to OneNet
sensorData ← getSensorData()
Socket → sendToOneNet(sensorData)
```

```
3 Thread2: RPi Receives command from OneNet
receivePackage ← Socket.receiveFromOneNet()
Command ← getCommand(receivePackage)
if Command is 'on' then
    onDevice()
end if
if Command is 'off' then
    offDevice()
end if
```

그림 5. OneNet의 의사 코드
Fig. 5 Pseudocode of OneNet

The device sends a CONN_REQ setup request, and the platform responds with CONN_RESP to establish a connection. The device sends PUSH_DATA data to OneNet. OneNet forwards the App message according to the destination address, and the App sends a PUSH_DATA control command to the OneNet. OneNet delivers the message to the device based on the destination address. The pseudocode is shown in Fig.5.

### 3.2 Flask server in the local

Flask is composed of WSGI(Web Server Gateway Interface), Jinja2, and Flask core as shown in Fig.6[4]. WSGI is used for routing and handling HTTP requests and HTTP responses. Jinja2 is used for template rendering and changing the variable data of a web site. Flask core is used to coordinate WSGI and Jinja2, and log data is stored in the SQLite. Ajax is a technology that can update some web pages without reloading the entire web pages[13].

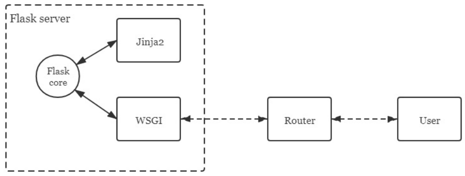


그림 6. 플래스크 서버
Fig. 6 Flask server

The user can view the data by accessing the web page refreshed by Ajax in real time. When the user clicks the button in App to send a command, the command is activated by a JavaScript click trigger event and forwarded to the Flask server for processing via Ajax. The user can use the phone to control the device and view its status through App in real time.

### 3.3 Video monitoring

The video monitoring framework is composed of RPi, camera, and cloud as shown in Fig.7. The camera is connected to the RPi through the CSI interface. The RPi obtains the video through the camera and sends it to the RTSP server running in the RPi. The user can access the RPi through the smartphone or PC to watch the video when the user is indoors, but the RPi cannot be directly accessed when the user is outdoors. Cloud supports video streaming, and RPi can transmit the video from the camera to the cloud, and the user can watch the video remotely from the cloud.

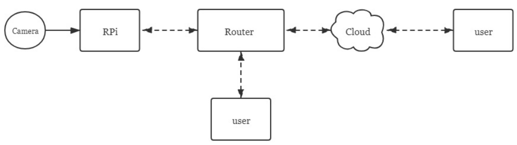


그림 7. 비디오 모니터링 프레임워크
Fig. 7 Video monitoring framework

The RTSP server consists of FFmpeg and Live555 as shown in Fig.8[14][15]. FFmpeg is used to get the video streaming from the camera, compress it into H264 format, and then send it to the Live555. The Live555 server is used to receive the video streaming of the H264 video format from FFmpeg and send the data of the H264 video format to the user using the RTSP protocol[14]. Before Live555 routes and distributes the video streaming, it needs to receive the RTSP request from the user.

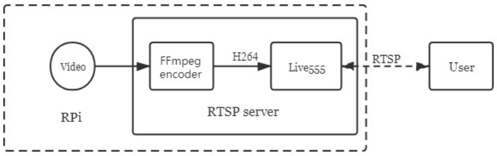


그림 8. RTSP 서버 프레임워크
Fig. 8 RTSP server framework

## IV. Experimental Results

The implemented home security system is shown in Fig.9. It has sensors such as temperature, humidity, motion, gas, fire, soil, rain, a camera, and actuators. All devices are connected to RPi, and the user can obtain data of sensors and control a relay and a motor.

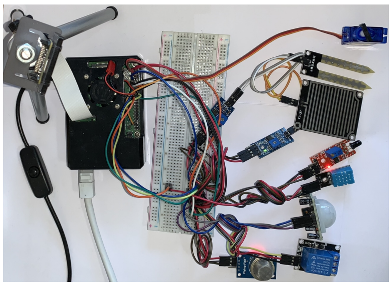

그림 9. 시스템의 구성
Fg. 9 Components of the system

User can access the home security system through two ways, OneNet cloud and Flask web server, depending on his location. User in the cloud can access OneNet through EDP protocol with two parameters: DEV_ID and API_KEY, and the user in the local can access Flask through HTTP.

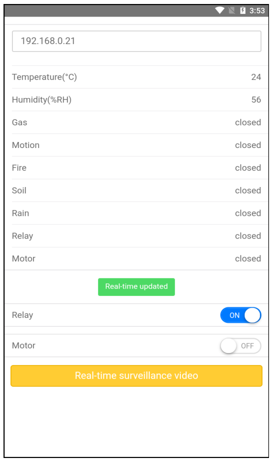


그림 10. 앱의 제어 판넬
Fig. 10 Control panel in App

The control panel in OneNet as shown in Fig.10 visualizes the received sensor data, and the user can turn on or off the relay and motor. The App sends ON/OFF command for devices to the RPi that switches the devices via the actuators.

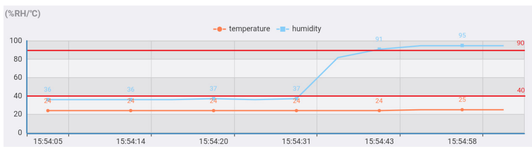


그림 11. OneNet의 대쉬보드
Fig. 11 Dash board in OneNet

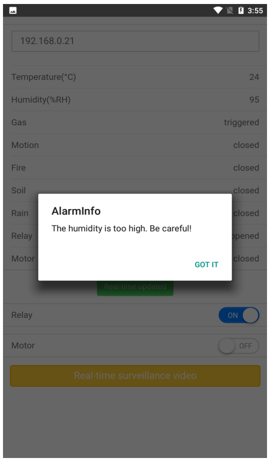


그림 12. 경고 메시지
Fig. 12 Warning message

The warning message is sent to the user as shown in Fig.12, when the temperature/humidity is over reference level(red line) as shown in Fig.11 water leak is detected, or a person is recognized. And the user can watch the video of the home through the smartphone as shown in Fig.13.

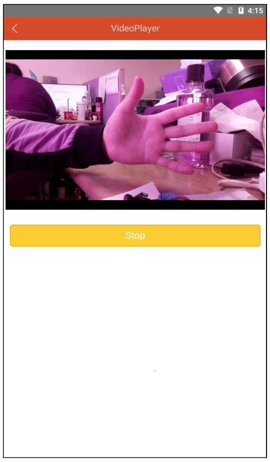


그림 13. 비디오스트림 스크린 캡쳐
Fig. 13 Screen capture of a video Stream

## V.Conclusion

This paper implements an IoT home security system that monitors the home in real time. RPi connects to OneNet through EDP protocol, and the user connects to the RPi remotely through OneNet. Flask provides local web server, and the user connects devices in a house and can view data on the smartphone and see real-time changes in data and status of sensors. When an abnormality is detected, an alarm message is sent to the user. User can see the video stream in real time and control the actuator through the App.

## References

[1] K. Park and D. Suh, "IoT Based Office Environment Improvement Plan," *J. of the Korea Institute of Electronic Communication Science,* vol. 15, no. 1, Feb. 29. 2020, pp. 61-70.

[2] K. Kim, D. Wang, and C. Han, "Home Security System Based on IoT," *J. of the Korea Institute of Electronic Communication Science,* vol. 12, no. 01, Feb. 2017, pp. 147-154.

[3] J. Du, J. Guo, D. Xu, and Q. Huang, "A remote monitoring system of temperature and humidity based on OneNet cloud service platform," *2017 IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS),* Haining, China, 2017, pp. 1-3.

[4] M. R. Mufid, A. Basofi, M. U. H. Al Rasyid, I. F. Rochimansyah, and A. rokhim, "Design an MVC Model using Python for Flask Framework Development," *2019 International Electronics Symposium (IES),* Surabaya, Indonesia, 2019, pp. 214-219.

[5] Z. Sun, S. Pan, and L. Yang, "Remote Monitoring System for Multiple Mobile Robot Based on OneNet Cloud Platform," *2018 Chinese Automation Congress (CAC),* Xi'an, China, 2018, pp. 3977-3980.

[6] D. Chu, C. Jiang, Z. Hao, and W. Jiang, "The

Design and Implementation of Video Surveillance System Based on H.264, SIP, RTP/RTCP and RTSP," *2013 Sixth International Symposium on Computational Intelligence and Design*, Hangzhou, China, 2013, pp. 39-43.

[7] R. Piyare and M. Tazil, "Bluetooth based home automation system using cell phone," *2011 IEEE 15th International Symposium on Consumer Electronics (ISCE)*, Singapore, Singapore, 2011, pp. 192-195.

[8] R. Teymourzadeh, Salah Addin Ahmed, Kok Wai Chan, and Mok Vee Hoong, "Smart GSM based Home Automation System," *2013 IEEE Conference on Systems, Process & Control (ICSPC)*, Kuala Lumpur, Malaysia, 2013, pp. 306-309.

[9] D. Sunehra and M. Veena, "Implementation of interactive home automation systems based on email and Bluetooth technologies," *2015 International Conference on Information Processing (ICIP)*, Pune, India, 2015, pp. 458-463.

[10] J. Du, J. Guo, D. Xu, and Q. Huang, "A remote monitoring system of temperature and humidity based on OneNet cloud service platform," *2017 IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS)*, Haining, China, 2017, pp. 1-3.

[11] S. Jindarat and P. Wuttidittachotti, "Smart farm monitoring using Raspberry Pi and Arduino," *2015 International Conference on Computer, Communications, and Control Technology (I4CT)*, Kuching, Malaysia, 2015, pp. 284-288.

[12] U. Sakthivel, N. Singhal, and P. Raj, "RESTful web services composition & performance evaluation with different databases," *2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, Mysuru, India, 2017, pp. 1-4.

[13] M. R. Mufid, A. Basofi, M. U. H. Al Rasyid, I. F. Rochimansyah, and A. rokhim, "Design an MVC Model using Python for Flask Framework Development," *2019 International Electronics Symposium (IES)*, Surabaya, Indonesia, 2019, pp. 214-219.

[14] Y. Tashtoush, M. N. AlRashdan, O. Salameh, and M. Alsmirat, "Swagger-based jQuery Ajax Validation," *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 2019, pp. 0069-0072.

[15] Y. Lee, J. Cho and C. Kim, "Efficient Authentication for Convergence of IoT and Mobile IP," *J. of the Korea Institute of Electronic Communication Science*, vol. 9, no. 6, Jun. 28, 2019, pp. 13-18.

## 저자 소개

**김강철(Kang-Chul Kim)**

1981년 서강대학교 전자공학과 학사
1983년 서강대학교 전자공학과 석사
1996년 경상대학교 전자공학과 박사
현재 전남대학교 전기전자통신컴퓨터공학부 교수
※ 관심분야 : 임베디드시스템, NoC, IoT Pattern Recognition

**Yongjiang Zhao**

2018년. Beijing Institute of Petrochemical Technology, Communication Engineering 졸업(공학사)
2020. 현재 전남대학교 대학원 컴퓨터공학과 재학
※ 관심분야 : Security, IoT