Performance Evaluation of a Smart CoAP Gateway for Remote Home Safety Services

Hyun-Sik Kim[1], Jong-Su Seo[1] and Jeongwook Seo[2]
[1] School of Electrical and Electronic Engineering, Yonsei University
Seoul, Korea
[e-mail: hskim@keti.re.kr, jsseo@yonsei.ac.kr]
[2] Department of Information and Communication Engineering, Namseoul University
Cheonan, Korea
[e-mail: jwseo@nsu.ac.kr]
*Corresponding author: Jeongwook Seo

Received April 14, 2015; revised July 28, 2015; accepted August 5, 2015;
published August 31, 2015

Abstract

In this paper, a smart constrained application protocol (CoAP)-based gateway with a border router is proposed for home safety services to remotely monitor the trespass, fire, and indoor air quality. The smart CoAP gateway controls a home safety sensor node with a pyroelectric infrared motion sensor, a fire sensor, a humidity and temperature sensor, and a non-dispersive infrared CO2 sensor and gathers sensing data from them. In addition, it can convert physical sensing data into understandable information and perform packet conversion as a border router for seamless connection between a low-power wireless personal area network (6LoWPAN) and the Internet (IPv6). Implementation and laboratory test results verify the feasibility of the smart CoAP gateway which especially can provide about 97.20% data throughput.

Keywords: Home satety, CoAP, gateway, border router, wireless connectivity

A preliminary version of this paper was presented at ICONI 2014, and was selected as an outstanding paper. This version includes the smart CoAP gateway as well as the home safety sensors previously dealt with and proivdes their implementation and laboratory results. This work was supported by the IT R&D program (World Class 300 Project) of MOTIE/KEIT, the Korean government [10046938, Cloud Services enabling UHD Media Platform for Video Transmission Based on Next-generation Wireless Communications at Indoor/Outdoor Area].

http://dx.doi.org/10.3837/tiis.2015.08.019 **ISSN : 1976-7277**

1. Introduction

Moden people spend over 80% of time indoors, and especially the elderly, housewives, and infants do almost all the time. However, due to the use of building materials such as insulating material in consideration of energy saving, a variety of contaminants are emitted and they cause serious health problems in people [1]. Also, people are sometimes exposed to dangerous situations such as an unlawful entry into a house, a fire, etc. Therefore, the monitoring and management of indoor environments such as public-use facilities, residential facilities, and child-care facilities becomes an important social issue from the viewpoint of home safety services which may not be provided by a few lax regulations and policies.

From a technical perspective on home safety problems, many possible solutions related to machine-to-machine (M2M) and internet-of-things (IoT) have been investigated in [2-5]. In [2], the requirements for M2M platforms with the considerations about IoT technologies was investigated. The authors in [3] presented the interoperablility at application layer by using representational state transfer (REST) principles. The benchmarking methods for IoT devices were presented in [4], and service discovery protocols for constrained M2M communications were introduced in [5]. Based on these literatures, the constrained application protocol (CoAP) is considered as one of the most promising technologies for the M2M and IoT devices. Thus, the CoAP has also been studied academically in [6-12]. A system architecture for IoT cloud services based on the CoAP was presented in [6], and a common middleware supporting message queue telemetry transport (MQTT) and CoAP was considered in [7]. A basic design concept to exploiting the CoAP for home automation was introduced in [8]. The CoAP was implemented on operating systems such as TinyOS in [9] and Contiki in [10]. Challenges was addressed for controlling interactions with constrained networks related to the public Internet in [11]. Compared to hypertext transfer protocol (HTTP), the performance of the CoAP was presented in terms of mote's energy consumption and response time in [12].

In this paper, therefore, we present a novel and practical solution to provide home safety services which can be realized by the proposed smart CoAP gateway and home safety sensor node. The home safety sensor node will be designed to support multiple sensor according to their applications, and the smart CoAP gateway with a border router will be designed to support multiple wireless connectivity and use the CoAP to communicate interactively and seamlessly between a low-power wireless personal area network and the Internet.

In Section 2, the proposed smart CoAP gateway and home safety sensor node are described in detail where hardware and software design methods are explained. Section 3 describes the laboratory test environment and scenario and presents some implementation and experimental results to evaluate the performance of the proposed smart CoAP gateway in cooperation with the home safety sensor node. Finally, some concluding remarks are given in Section 4.

2. Smart CoAP Gateway and Home Safety Sensor Node

2.1 Design of Home Safety Sensor Node

In this section, the proposed home safety sensor (HSS) node is described. The HSS node is designed to support multiple sensors such as a pyroelectric infrared (PIR) motion sensor, a non-dispersive infrared (NDIR) CO_2 sensor, a fire sensor, and a temperature/humidity sensor as shown in **Fig. 1**. However, it does not have any communication module but only has a

RS232 interface to a CoAP device with communication modules which uses the same hardware as a smart CoAP gateway described in the next section. Thus, various HSS nodes with other sensor applications can be applied to the same CoAP device. The PIR motion sensor can detect a small movement within the range of 5 meters by using quad-type pyroelectric element. The fire sensor can detect the smoke within the range of 0.3~5%/foot in 3 seconds by using an ionization smoke chamber. The temperature and humidity sensor is able to measure temperature with the range of -40~123.8°C and humidity with the range of 0~100% RH. The NDIR CO_2 sensor can measure CO_2 concentration levels up to 2000 and 5000 ppm.

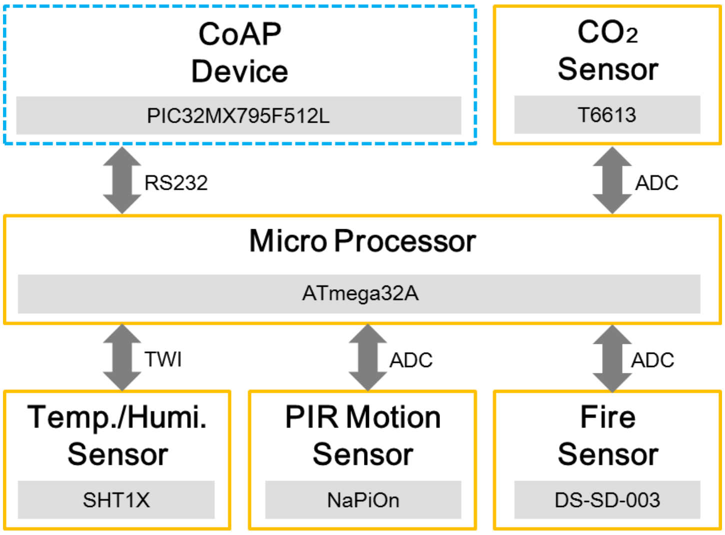


Fig. 1. Block diagram of the proposed home safety sensor (HSS) node

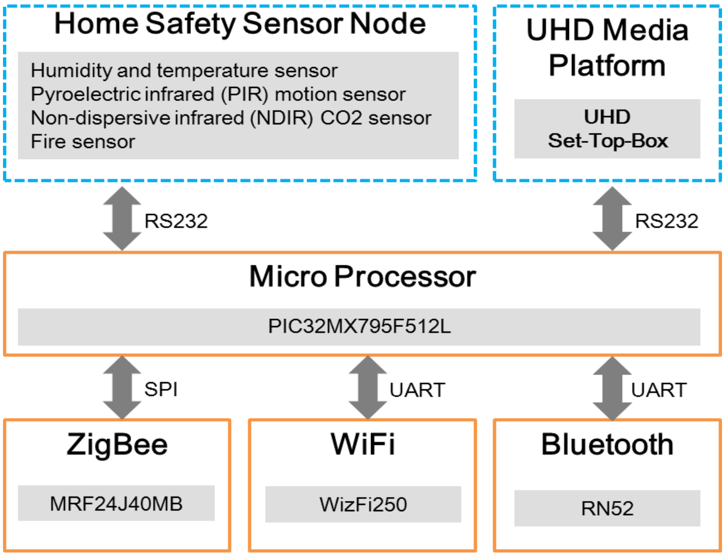


Fig. 2. Block diagram of the smart CoAP gateway

2.2 Design of Smart CoAP Gateway with a Border Router

The proposed smart CoAP gateway is illustrated in **Fig. 2**. It provides multiple connectivity for communicating with smart devices and sensor nodes through WiFi, Bluetooth, and Zigbee. It has RS232 interfaces of the HSS node and the ultra-high definition (UHD) media platform. The RS232 interface to the UHD media platform (namely, UHD set-top box) enables the smart CoAP gateway to be included in set-top boxes.

The software architecture of the smart CoAP gateway is shown in **Fig. 3**. It is implemented on Contiki operating system (OS) with support for dynamic loading and replacement of individual programs and services in [13]. The Contiki aims to provide a light-weight operating system for resource-contrained devices. It contains various useful components such as timers, radio drivers, radio duty cycling drivers, threads and processes. Also, it can easily add and modify components in a scalable way. The CoAP denotes a specialized web transfer protocol to be used with constrained nodes which usually have only 8-bit microcontrollers with limited ROM and RAM and constrained networks which may have high packet error rates and a typical throughput of 10s of kbps. It is similar to the well-known HTTP.

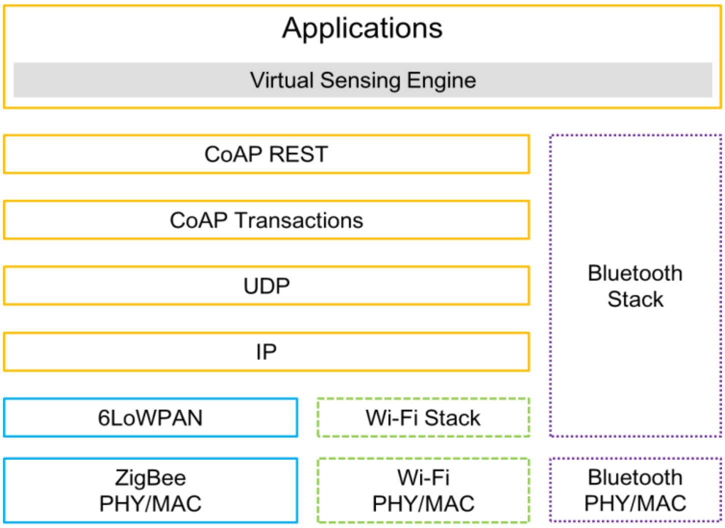


Fig. 3. Software architecture of the smart CoAP gateway

Both CoAP and HTTP use uniform resource identifiers (URIs) to locate resources, and they also shares a common set of request methods: GET, POST, PUT and DELETE in **Table 1**.

Table 1. Description of the CoAP methods

Method	Description
GET	Retrieves information of an identified resource
POST	Creates a new resource under the requested URI
PUT	Updates the resource identified by an URI
DELETE	Deletes the resource identified by an URI

Table 2. Description of the CoAP messages

Message	Description
CON	Confirmable requests that the receiving peer sends an acknowledgement or a reset
NON	Non-confirmable messages do not request any message being sent by the receiving peer
ACK	Acknowledges that a CON has been received, may carry payload
RST	Indicates that a CON has been received but some context is missing to process it

These methods makes the CoAP easy to integrate into the current web. In order to make the CoAP especially suited for contrained nodes, it has some important differences compared to HTTP. First of all, the CoAP does not require a reliable transport protocol like transmission control protocol (TCP) since the reliable tranport protocol increases the complexity, size and resource usage of the software, which may be unwanted or even impossible to use on constrained nodes. Thus, it uses user datagram protocol (UDP) and implements its own optional, light-weight, and simple reliability mechanism. Another difference is the format of the header. In HTTP options and request method are transmitted in clear-text, which means that even a basic request consists of several bytes of data. Instead, the CoAP header has a binary format where request method and options are encoded into various bits and in a specific order. This reduces the data which has to be sent, received and parsed by the endpoints.

The CoAP is designed to realize the REST architecture for machine-to-machine (M2M) or internet-of-things (IoT) applications. It has the main feature: 1) web protocol fulfilling M2M requirements in constrained environments, 2) UDP binding with optional reliability supporting unicast and multicast requests, 3) asynchronous message exchange, 4) low header overhead and parsing complexity, 5) URI and content-type support, 6) simple proxy and caching capabilities, 7) a stateless HTTP mapping, allowing proxies to be built providing access to CoAP resources via HTTP in a uniform way or for HTTP simple interfaces to be realized alternatively over CoAP, 8) Security binding to datagram transport layer security (DTLS). The interaction model of the CoAP is similar to the client and server model of the HTTP, but M2M/IoT interactions typically requires the CoAP acting in both client and server roles. A CoAP request is equivalent to that of HTTP. It is sent by a client to request an action using a method code on a resource identified by a URI on a server. Then, the server sends a response with a response code.

Note that the CoAP handles these interchanges asynchronously over a datagram oriented transport. In other words, its messaging model is based on the messages exchange over UDP as mentioned above. In [14], its message format has a four bytes binary header which may be followed by compact binary options and a payload, and it is shared by requests and responses. Each message contains a message identifier (ID) for in order to detect duplicates and for optional reliability. The CoAP defines four types of messages such as Confirmable (CON), Non-confirmable (NON), Acknowledgement (ACK), and Reset (RST) as shown in **Table 2**. In order to provide reliability, the CON message is used, that may be retransmitted by a default timeout and exponential back-off between retransmissions until the recipient sends an ACK message with the same message ID from the corresponding endpoint. Then, if a recipient cannot process a CON message, it replies with a RST message instead of an ACK message. A message not requiring reliability can be sent as a NON message. Although the NON message does not require an ACK, it still has a message ID for duplicate detection. When a recipient cannot process a NON message, it may reply with a RST message. As an example of the CoAP request and response model, a basic GET request with piggybacked response is shown in Fig. 4 where a token is used to match responses to requests independently from the underlying messages.

In Fig. 5, the network encapsulation and decapsulation procedure related to the CoAP is illustrated. When the CoAP data such as a HSS sensor data request moves from upper layer to lower level of the protocol stack where each layer includes a bundle of relevant information called a header along with the actual data. The data package containing the header and the data from the upper layer then becomes the data repackaged at the next lower level with lower layer's header. The header is used at the receiving side to extract the data from the encapsulated data packet. This packing of data at each layer is known as encapsulation. Decapsulation is the reverse process of encapsulation, which occurs when data is received on the other endpoint. As the data moves up from the lower layer to the upper layer of the protocol stack, each layer unpacks the corresponding header and uses the information contained in the header to deliver the packet to the exact network application waiting for the data.

As an application between the smart CoAP gateway and the HSS node, a virtual sensing engine such as [15] can be implemented to convert physical sensing data into information easily understood by non-experts. For instance, the data from a PIR motion sensor can be converted into a warning message such that a trespass has occurred.

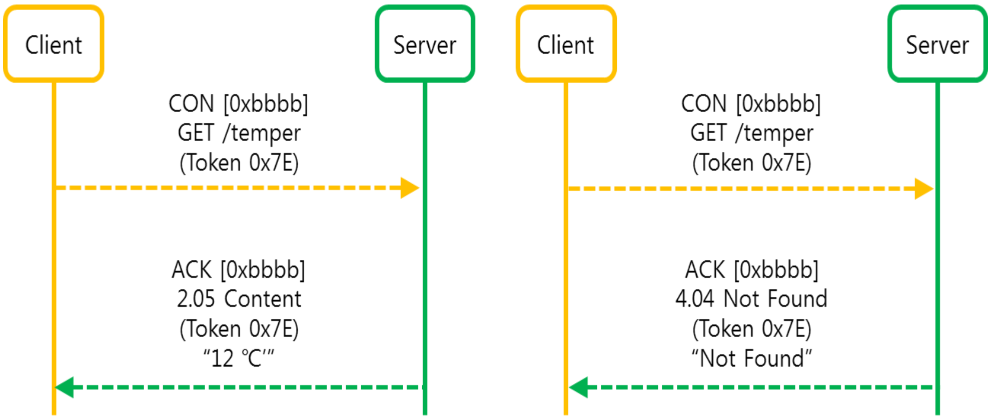


Fig. 4. Example of the CoAP interaction model : a basic GET resquest with piggybacked response

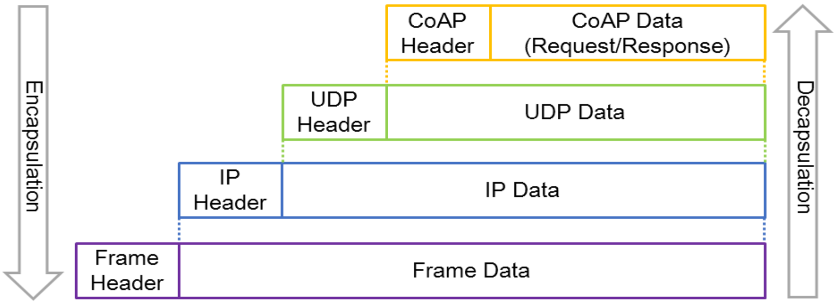


Fig. 5. Network encapsulation and decapsulation procedure

The smart CoAP gateway supports a border router to seamlessly connect the HSS node in a low-power wireless personal area network (6LoWPAN) to the Internet (IPv6) as shown in **Fig. 6**. The border router has a routing function and a serial line over IP (SLIP) function used to encapsulate IP packets and send them over a serial link as shown in **Fig. 7**. It can translate between IPv6 and 6LoWPAN where IPv6 is used on the serial line (Ethernet) and 6LoWPAN is used on the wireless link (IEEE 802.15.4).

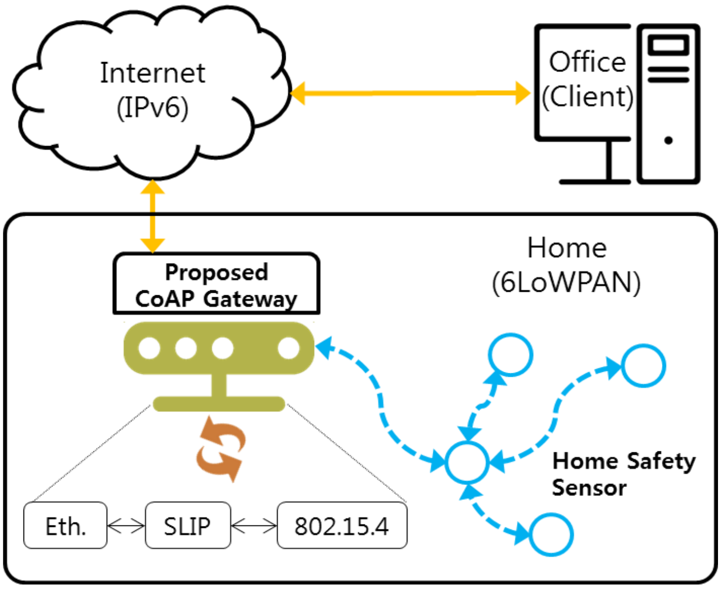


Fig. 6. Border router for seamless connection between 6LoWPAN and Internet

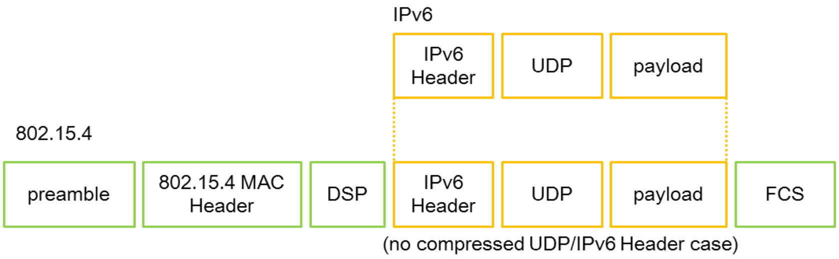


Fig. 7. Packet conversion between IPv6 and 6LoWPAN

3. Implementation and Laboratory Test Results

In order to verify the feasibility of the smart CoAP gateway with a border router and the HSS node, implementation and laboratory results are shown in this section.

Fig. 8 illustrates the Contiki simulation result to show the CoAP usability where its data volumes required to transmit two messages are compared to HTTP/TCP and HTTP/UDP. The HTTP/TCP protocol spends the largest data volume because it sends a lot of control data in the transport layer, whereas the CoAP/UDP does the smallest one. In addition, the CoAP/UDP outperforms the HTTP/UDP in the application layer.

Fig. 9 illustrates the implemetation results of the smart CoAP gateway and the HSS node. The smart CoAP gateway includes multiple communication modules such as Wi-Fi, Zigbee and Bluetooth while the HSS node includes multiple sensors such as PIR motion, fire, temperature/humidity, and CO_2. They can be connected by a RS232 interface.

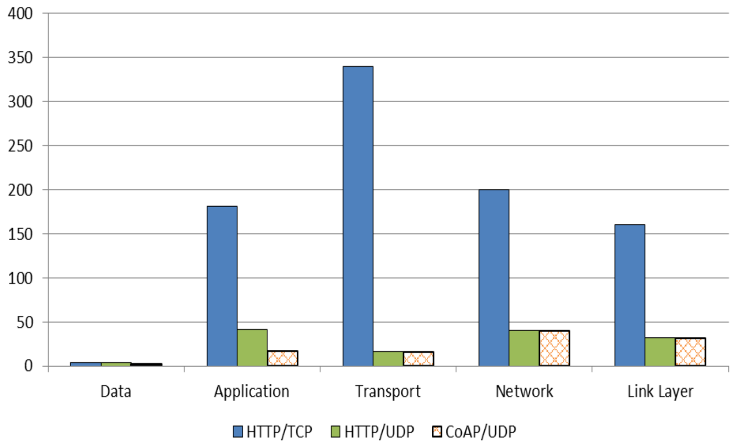
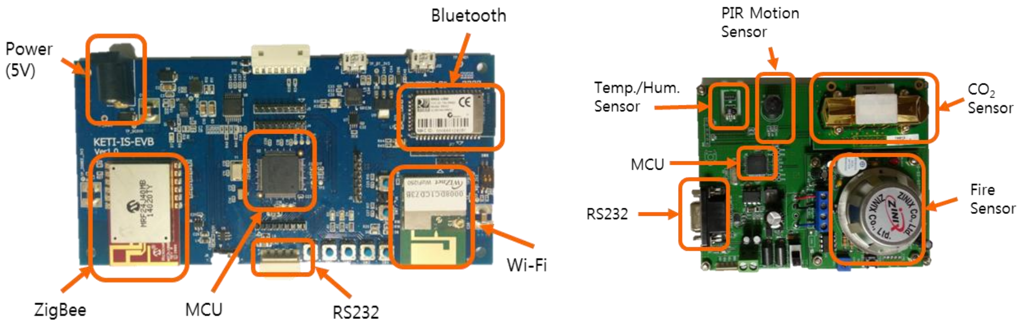


Fig. 8. Comparison of data volumes between CoAP/UDP, HTTP/TCP and HTTP/UDP



(a) Smart CoAP gateway (b) HSS node with multiple sensors

Fig. 9. Implementation results of the proposed CoAP gateway and HSS node

Fig. 10 illustrates the laboratory test environment and scenario to verify the remote connection of the smart CoAP gateway with a border router. The laboratory test environment represents the same concept shown in **Fig. 6**. Through a smart CoAP gateway, a client in his office performs a GET request and a server in a CoAP device, which is physically connected to the HSS node by a RS232 interface, performs a corresponding response. The client is connected to a Hub via an Ethernet cable, and the Ethernet Hub is connected to the smart CoAP gateway via an Ethernet cable. The smart CoAP gateway communicates with the CoAP device connected to the HSS node via ZigBee. For the client, a test program is developed by Microsoft Visual Studio 2013 where the client will send a request message (CON) and wait for the response message (ACK) to be arrived during 10 seconds. If the response message does not arrive from the server, the client will wait in 93 milliseconds and the performs the retransmission.

Fig. 11 shows the labrotory test results where the test was repeated thirty times. The client peformed total 3760 requests and successfully received the 3655 responses. Therefore, the data throughput of the implemented smart CoAP gateway and the HSS node is approximately 97.20%. In general, data throughput varies depending on the real network environment and its traffic conditions. Nevertheless, the performance obtained is comparable to the result in [16] which showed the performance of more than 95% under different experimental conditions. Moreover, the smart CoAP gateway can work well with smart phones/pads through Bluetooth and WiFi.

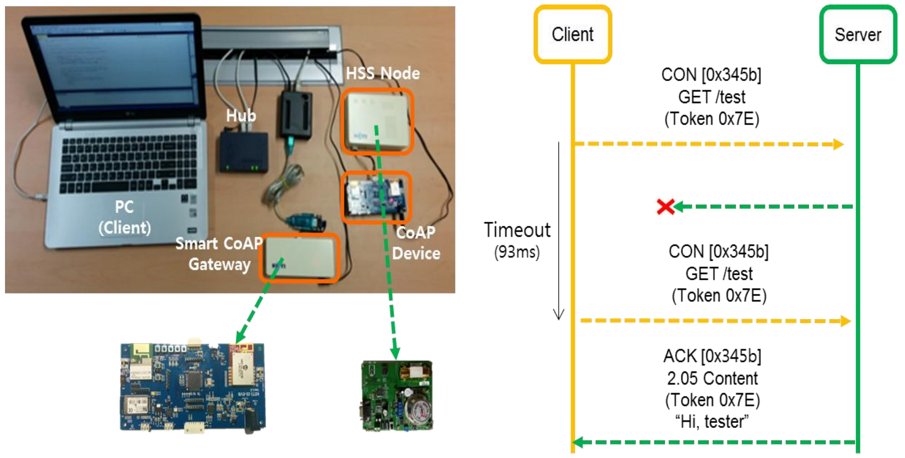
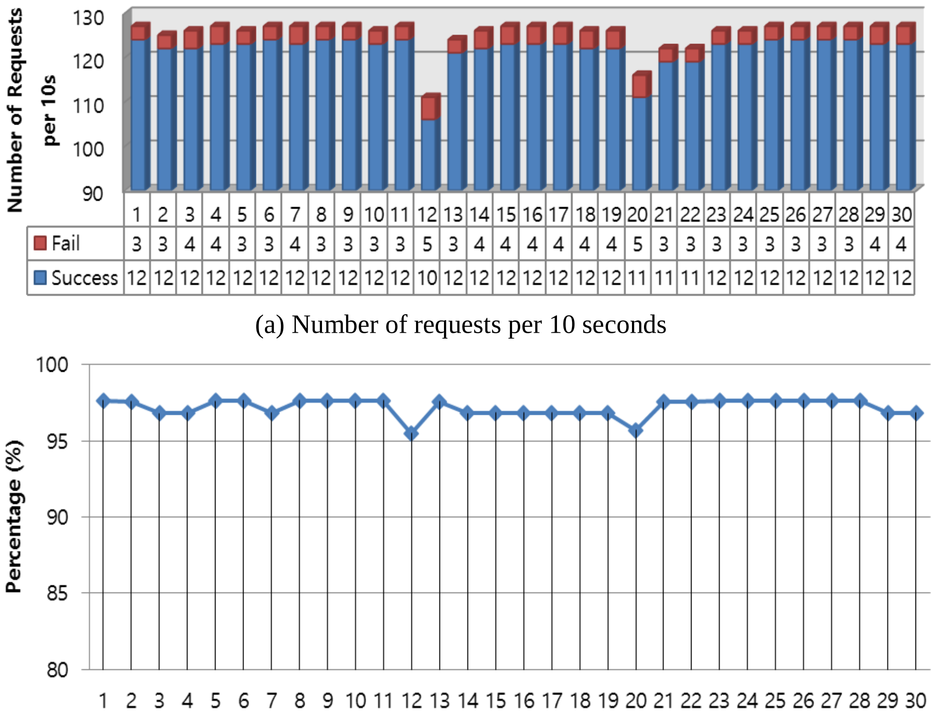


Fig. 10. Laboratory test environment and scenario



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
🟥 Fail	3	3	4	4	3	3	4	3	3	3	3	5	3	4	4	4	4	4	4	5	3	3	3	3	3	3	3	3	4	4
🟦 Success	12	12	12	12	12	12	12	12	12	12	12	10	12	12	12	12	12	12	12	11	11	11	12	12	12	12	12	12	12	12

(a) Number of requests per 10 seconds

(a) Data throughput

Fig. 11. Laboratory test results : number of requests and data throughput

4. Conclusion

In this paper, a smart CoAP gateway with a border router and the home safety sensor node were designed and implemented to provide home safety services which can be controlled and monitored by a client on a remote location. Basically, the smart CoAP gateway is able to communicate with smart devices such as smart phones and pads via Bluetooth and WiFi and sensor nodes via ZigBee. The home safe sensor node was designed to support multiple sensors such as a PIR motion, a temperature and humidity, an NDIR CO2 sensor, and a fire sensors. It can be connected to a CoAP device or a smart CoAP gateway via a RS232 interface. The smart CoAP gateway can convert physical sensing data into the information easily understandable via the virtual sensing engine. Moreover, thanks to a border router in the smart CoAP gateway, it is possible to communicate interactively over the Internet. In other words, the seamless connection between a low-power wireless personal area network and the Internet is provided. Some laboratory test results evaluate the performance of the proposed smart CoAP gateway which can provide about 97.20% data throughput.

References

[1] United States Environmental Protection Agency, *What are the trends in indoor air quality and their effects on human health*, Report on the Environment, 2011. Article (CrossRef Link)

[2] M. Castro, A. J. Jara, and A. F. Skarmeta, "An analysis of M2M platforms: challenges and opportunities for the Internet of Things," in *Proc. of 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, pp. 757-762, July 2012. Article (CrossRef Link)

[3] R. V. Chander, et al., "A REST based design for Web of Things in smart environments," in *Proc. of 2012 2nd IEEE International Conference on Parallel Distributed and Grid Computing (PDGC)*, pp. 337-342, December 2012. Article (CrossRef Link)

[4] C. P. Kruger and G. P. Hancke, "Benchmarking Internet of things devices," in *Proc. of 2014 12th IEEE International Conference on Industrial Informatics (INDIN)*, pp. 611-616, July 2014. Article (CrossRef Link)

[5] B. C. Villaverde, et al., "Service Discovery Protocols for Constrained Machine-to-Machine Communications," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 41-60, 2014. Article (CrossRef Link)

[6] M. Kovatsch, L. Martin, and Z. Shelby, "Californium: Scalable cloud services for the internet of things with coap," in *Proc. of the 4th International Conference on the Internet of Things (IoT 2014)*, pp. 1-6, October 2014. Article (CrossRef Link)

[7] D. Thangavel, et al., "Performance evaluation of MQTT and CoAP via a common middleware," in *Proc. of 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pp. 1-6, April 2014. Article (CrossRef Link)

[8] O. Bergmann, K.T. Hillmann, and S. Gerdes, "A CoAP-gateway for smart homes," in *Proc. of International Conference on Computing, Networking and Communications (ICNC)*, pp. 446-450, Feb. 2012. Article (CrossRef Link)

[9] T. Potsch, et al., "Performance evaluation of CoAP using RPL and LPL in TinyOS," in *Proc. of 2012 5th International Conference on New Technologies, Mobility and Security (NTMS)*, pp. 1-5, May 2012. Article (CrossRef Link)

[10] M. Kovatsch, S. Duquennoy, and D. Adam, "A low-power CoAP for Contiki," in *Proc. of 2011 IEEE 8th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, pp. 855-860, October 2011. Article (CrossRef Link)

[11] F. Van den Abeele, et al., "Fine-grained management of CoAP interactions with constrained IoT devices," in *Proc. of Network Operations and Management Symposium (NOMS)*, pp. 1-5, May 2014. Article (CrossRef Link)

[12] W. Colitti, et al., "Evaluation of constrained application protocol for wireless sensor networks," in *Proc. of 2011 18th IEEE Workshop on Local & Metropolitan Area Networks (LANMAN)*, pp. 1-6, October 2011. Article (CrossRef Link)

[13] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki – a lightweight and flexible operating system for tiny networked sensors," in *Proc. of International Conference on Local Computer Networks (LCN)*, pp. 16-18, November 2004. Article (CrossRef Link)

[14] Z. Shelby, K. Hartke, and C. Bormann, *The Constrained Application Protocol (CoAP)*, IETF RFC 7252, June, 2014. Article (CrossRef Link)

[15] L. Liu, S. M. Kuo, and M. Zhou, "Virtual sensing technologies and their applications," in *Proc. of International Conference on Networking, Sensing and Control*, pp. 31-36, March 2009. Article (CrossRef Link)

[16] V. Varga, L. Kriara, V. Vukadinovic, T. Gross, and S. Mangold, "Demo: bringing the Internet of Things of toys to life," in *Proc. of International Conference on Mobile Systems, Applications, and Services*, pp. 461, May 2015. Article (CrossRef Link)

Hyunsik Kim received the BS and MS degrees in information and communication engineering from the Inah University, South Korea, in 2002 and 2004, respectively. Currently, he is a Ph.D. candidate in the electrical & electronic engineering from the Yonsei University, South Korea. He is also working as a senior researcher in the Contents Convergence Research Center at the Korea Electronics Technology Institute (KETI), South Korea from 2004. His research interests are in contents sharing, wearable computing, human-computer interaction, and Machine-to-Machine/Internet of Things.

Jong-Soo Seo received the B.S. degree in electronics engineering from Yonsei University, Seoul, Korea, in 1975, and the M.S. and Ph.D. degrees from the University of Ottawa, Ottawa, ON, Canada, in 1983 and 1988, respectively. He was with IDC and CAL, Canada, engaged in research on digital satellite communications and data broadcasting systems for six years. Since 1995, he has been with the Department of Electrical and Electronic Engineering, Yonsei University, where he is currently a Professor. Prof. Seo is an IEEE Fellow and an Associate Editor of the IEEE Transactions on Broadcasting. His current research interests include next generation broadcasting and 5G radio systems.

Jeongwook Seo received the B.S. and M.S. degrees from the Department of Telecommunication and Information Engineering, Korea Aerospace University, Gyeonggi, Korea, in 1999 and 2001, respectively, and the Ph.D. degree from the School of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea, in 2010. He is currently an Assistant Professor at the Department of Information and Communication Engineering, Namseoul University, Cheonan, Korea. From 2001 to 2013, he was with Network Convergence Research Center in Korea Electronics Technology Institute, Seoul, Korea. His research interests include statistical signal processing for communications and networking and protocol design for Machine-to-Machine/Internet of Things.