

특집논문 (Special Paper)

방송공학회논문지 제21권 제3호, 2016년 5월 (JBE Vol. 21, No. 3, May 2016)

<http://dx.doi.org/10.5909/JBE.2016.21.3.298>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

BLE 네트워크 상에서 사물인터넷 서비스 제공을 위한 CoAP과 6LoWPAN 구현

김철민^{a)}, 강형우^{a)}, 최상일^{a)}, 고석주^{a)‡}

Implementation of CoAP/6LoWPAN over BLE Networks for IoT Services

Cheol-Min Kim^{a)}, Hyung-Woo Kang^{a)}, Sang-Il Choi^{a)}, and Seok-Joo Koh^{a)‡}

요 약

최근 사물인터넷 기술의 등장과 함께 소형 장치를 이용한 스마트 홈 및 헬스케어 서비스 등의 다양한 연구가 진행되고 있다. 기존 사물 간 통신은 인터넷 통신을 사용하지 않고 Machine to Machine (M2M) 기반의 블루투스 및 ZigBee와 같이 하위 계층의 Personal Area Network (PAN) 기술을 이용하므로 네트워크 장치간의 통신만 지원할 뿐, PAN 네트워크에 속하지 않은 장치와의 통신에 어려움이 많았다. 본 논문에서는 기존의 M2M 통신인 Bluetooth Low Energy (BLE) 네트워크에서 사물인터넷 서비스 제공을 위해 IPv6 over Low Power Wireless Personal Area Networks (6LoWPAN)을 구현하여 저사양 장치에서 인터넷 통신을 가능하도록 하였다. 이를 위해 사물인터넷의 핵심 프로토콜인 Constrained Application Protocol (CoAP)을 구현하였다. 테스트베드 구현 및 성능분석 실험을 통해 기존 Hypertext Transfer Protocol (HTTP)과 통신 성능을 비교한 결과, CoAP이 HTTP보다 약 2배 우수한 처리량, 약 21% 빠른 평균 전송시간 및 22% 적은 패킷 전송량을 보여줌을 알 수 있었다.

Abstract

With the advent of Internet of Things (IoT) technology that allows the communications between things and devices over the Internet, a lot of researches on the IoT services, such as smart home or healthcare, have been progressed. In the existing machine-to-machine (M2M) communications, however, since the underlying link-layer technologies, such as Bluetooth or ZigBee, do not use the Internet Protocol (IP) communication, those technologies are not suitable to provide the IoT services. Accordingly, this paper discusses how to provide the Internet services in the M2M communication, and propose an implementation of the Constrained Application Protocol (CoAP) over 6LoWPAN for providing IoT services in the BLE networks. Based on the implementation, we compared the performance between HTTP and CoAP for IoT communications. From the experimental results, we can see that the CoAP protocol gives better performance than the HTTP protocol with two times higher throughput, 21% faster transmission time, and 22% smaller amount of generated packets.

Keyword : IoT, Bluetooth Low Energy, 6LoWPAN, CoAP

a) 경북대학교 컴퓨터학부(School of Computer Science and Engineering, Kyungpook National University)

‡ Corresponding Author : 고석주(Seok-Joo Koh)

E-mail: sjkoh@knu.ac.kr

Tel: +82 53-950-7356

ORCID: <http://orcid.org/0000-0003-3429-2040>

※ 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 SW중심대학지원사업의 연구결과로 수행되었음.

※ 이 논문의 결과 중 일부는 "IWAIT 2016"에서 발표한 바 있음.

· Manuscript received March 3, 2016; Revised April 11, 2016; Accepted April 22, 2016.

1. 서론

최근 사물인터넷 기술이 이슈가 되면서 스마트 홈, 스마트 시티, 헬스케어 및 스마트 카 등 다양한 분야에서 연구가 활발히 진행되고 있다^[1]. 대표적으로 스마트 홈 분야의 동작 감지 기술을 이용한 실시간 감시 시스템^[2], 스마트 시티 분야의 무선 센서 네트워크 (Wireless Sensor Network, WSN)와 자동차 네트워크 (Car Area Network, CAN)를 이용한 실시간 이상 감시 시스템^[3], 그리고 스마트 운송 분야에서 사물인터넷을 이용한 실시간 모니터링 시스템 등의 응용이 있다^[4].

이러한 사물인터넷 기반의 서비스를 제공하기 위해, 구글, 애플, 네스트 (Nest Labs), 아마존, 삼성, LG 등의 국내 외의 여러 회사에서도 사물인터넷 기반의 기술에 대한 연구 및 투자를 아끼지 않고 있다. 또한, 구글의 구글 글라스 (Google Glass)와 나이키 (Nike)의 퓨얼밴드 (Fuel Band) 등 웨어러블 장치 (wearable device) 들의 발전과 초소형 장치들의 성능은 점점 향상되고 가격은 저렴해지면서 사물인터넷의 발전에 큰 영향을 미치고 있다.

하지만 대부분의 웨어러블 장치는 통신할 때 인터넷 프로토콜 (Internet Protocol, IP) 대신 ZigBee나 블루투스과 같은 링크 계층의 기술을 사용한다. M2M 통신 기반의 이러한 기술들은 PAN 네트워크 안의 사물 간 직접적인 통신만을 지원하기 때문에 PAN 네트워크에 속하지 않은 장치와 통신이 불가능하며, 이러한 이유로 인터넷을 이용한 통신 기반의 사물인터넷 서비스를 제공하는 데 적합하지 않다.

본 논문에서는 기존의 M2M 통신인 BLE 통신에서 6LoWPAN과 CoAP의 구현을 통해 사물인터넷 서비스 제공을 위한 통신기술을 구현하고 그 성능을 분석해보고자 한다. 6LoWPAN을 사용함으로써 장치는 IP 프로토콜을 이용할 수 있고 인터넷에 직접 연결될 수 있다^[5]. 또한, 기존에 활용되고 있는 HTTP 프로토콜은 사물인터넷에서 사용되는 장치들을 고려하지 않고 설계되었기 때문에, 텍스트 기반의 HTTP 헤더와 Transmission Control Protocol (TCP) 을 이용한 통신은 사물인터넷에서 사용되는 장치에서 이용하기에 많은 처리능력을 요구하며 통신에 필요한 전력소모가 많다. 이러한 문제를 해결하기 위해 사물인터넷에 사용

되는 저 사양 장치에 맞는 응용 프로토콜의 연구가 활발하게 진행되고 있다. 특히 IETF (Internet Engineering Task Force)의 CoRE (Constrained RESTful Environments) 워킹 그룹에서는 저 사양 장치에서의 통신을 위한 CoAP 프로토콜을 제안한다. CoAP은 HTTP에 비해 가볍고 통신에 사용되는 전력 소모량도 적은, 임베디드 장치에 최적화된 응용 프로토콜로써 사물인터넷 환경의 장치에서 HTTP 등의 기존 프로토콜보다 효율적으로 사용될 수 있다^[6].

본 논문의 구성은 다음과 같다. 2절에서는 6LoWPAN과 CoAP 등 관련된 연구를 소개한다. 3절에서는 사물인터넷 서비스를 제공하기 위한 BLE 네트워크에서의 6LoWPAN 및 CoAP의 구현에 대하여 설명할 것이다. 4절에서는 구현된 테스트베드에서의 실험을 통하여 BLE 네트워크에 구현한 6LoWPAN 환경의 CoAP과 기존 통신 기술인 HTTP의 통신 성능을 비교 분석할 것이다. 마지막으로 5절에서 본 논문의 결론을 짓는다.

II. 관련 연구 동향

1. 6LoWPAN

6LoWPAN은 IEEE 802.15.4를 PHY/MAC으로 하는 저 전력 무선 개인 네트워크 (Wireless Personal Area Network, WPAN) 상에서 IPv6를 탑재하여 기존 IP 네트워크와 연결하는 네트워크 기술이다^[7]. IETF의 6LoWPAN 워킹 그룹에서 저전력 무선 사설 네트워크, 센서 네트워크 위에서 인터넷 프로토콜을 사용하기 위한 표준을 정의하고 있다.

Application	Application	
Transport	UDP	ICMP
Network	IPv6	
Data Link	6LoWPAN Adaption Layer IEEE 802.15.4 MAC	
Physical	IEEE 802.15.4 PHY	

그림 1. 6LoWPAN 프로토콜 스택
 Fig. 1. 6LoWPAN protocol stack

6LoWPAN은 저전력으로 동작하며, 250Kbps의 적은 대역폭을 사용하고 IPv6에 비해 작은 패킷 크기를 활용하기 때문에 저 사양의 초소형 장치에서 활용하기 적합하다. 그림 1은 6LoWPAN에서의 프로토콜 스택을 보여준다.

6LoWPAN을 장치에서 활용하려면 그림 1과 같이 IEEE 802.15.4 표준을 만족하는 하위 계층의 통신 기술이 필요하며, 이에 최적화된 상위 계층의 작업이 필요하다. 6LoWPAN은 IEEE 802.15.4에 최적화된 상위 계층이다.

6LoWPAN은 헤더 압축과 컨텍스트(context)를 포함하는 글로벌 주소(global address) 압축 그리고 IPv6 확장 헤더의 압축 및 User Datagram Protocol(UDP)와 경량화된 멀티캐스트 주소(compact multicast address) 압축을 이용하여 제한된 장치에서 최적화된 통신을 제공한다.

2. CoAP

CoAP은 저성능의 CPU를 가지면서 작은 용량의 RAM 및 ROM을 갖는 저 사양 장치에서 사용하기 위한 응용 프로토콜이다^[8]. CoAP은 IETF의 CoRE 워킹그룹에서 6LoWPAN 상의 응용 프로토콜로 표준화를 진행하고 있다. CoAP은 HTTP에 비해 가볍고 전력을 적게 소모하며 빠른 처리 속도를 제공한다. 이 프로토콜은 하위 계층으로 UDP를 사용하며 6LoWPAN 환경에서 동작한다는 차이점 이외에는 HTTP와 유사하다^{[9][10]}. CoAP은 서버-클라이언트(Server - Client) 통신 모델을 사용한다. 이것은 클라이언트가 서버에게 자원(resource) 요청 메시지를 보내면, 서버에서 이를 수신하고 자원을 응답 메시지로 응답한다. 그림

2는 HTTP와 CoAP의 프로토콜 스택을 보여준다.

CoAP은 두 개의 큰 모듈인 메시지 전달(messaging) 모듈과 요청/응답 모듈로 이루어져 있다. 메시지 전달 모듈은 신뢰성(reliability) 있는 메시지 전달과 비 신뢰성의 메시지 전달을 제공한다. 신뢰성 있는 메시지 전달에는 conformable(CON) 메시지를 활용한다. 모든 CON 메시지는 acknowledgement(ACK) 메시지로 응답해야 한다. 클라이언트가 CON 메시지를 전송한 뒤 ACK 메시지를 수신하지 못하면, 메시지 전송 중 오류가 발생했다고 판단하고 메시지를 서버로 재전송한다. Non-conformable(NON) 메시지는 모든 NON 요청 메시지에 대하여 ACK 메시지를 전송하지 않는다. 즉 NON 메시지는 중간에 손실된 메시지에 대한 확인을 할 수가 없다.

요청/응답 모듈은 piggy-backed response와 separate response 그리고 non-conformable response를 제공한다.

CoAP의 자원은 Representational State Transfer(REST) 아키텍처로 접근할 수 있다^[11]. REST 구조에는 GET, POST, PUT, DELETE 메소드(method)가 있다. 이들은 자원에 대해 각각 검색, 생성, 수정, 삭제하는 것을 의미한다. 이들을 이용하여 “메소드 coap://서버 주소/자원”의 형태로 요청을 전달한다. 예를 들어 서버에 연결된 온도 센서의 측정값을 얻어오고 싶을 때 “GET coap://서버 주소/temperature”의 형식으로 요청한다.

CoAP은 요청에 관한 결과로 응답 코드를 이용한다. 이들은 HTTP의 응답코드와 유사하다. 2.xx으로 시작하는 응답코드는 “성공”을 의미하고, 4.xx는 “클라이언트 오류”, 그리고 5.xx은 “서버 오류”에 해당한다.

HTTP	Application	CoAP
TCP	Transport	UDP
IPv4, IPv6	Network	IPv6
Ethernet MAC	Data Link	6LoWPAN Adaption Layer
Ethernet PHY	Physical	IEEE 802.15.4 MAC
		IEEE 802.15.4 PHY

그림 2. HTTP와 CoAP의 프로토콜 스택
Fig. 2. Protocol stack of HTTP and CoAP

CoAP 표준은 다양한 환경에 맞게 구현이 되어있으며 대표적으로 C 언어로 구현한 libcoap, JAVA 언어로 구현한 Californium 이외에도 Node.js 기반의 node-coap 등 다양한 언어를 기반으로 구현되어 있다. 그뿐만 아니라 스마트폰, 웹 브라우저 및 임베디드 장치용으로 구현된 CoAP도 존재한다.

III. CoAP / 6LoWPAN / BLE 통신구조 구현

블루투스에서 IP 통신을 위한 Internet Protocol Support Profile (IPSP)을 제공한다^[12]. 이 프로파일은 블루투스 4.1 버전 이상이 요구되며, 6LoWPAN은 리눅스 커널 3.18 버전부터 지원이 된다. 그림 3은 리눅스에서 6LoWPAN 모듈을 활성화 시키고 6LoWPAN 게이트웨이 (gateway)와 CoAP 서버 간 6LoWPAN 연결을 생성한 뒤, IP를 할당하고 패킷이 제대로 전달될 수 있도록 라우팅 설정을 하는 과정이다.

IV. 실험 및 성능분석

1. 실험환경

그림 4는 구현된 테스트베드 구조도를 보여준다. 네트워크 A는 6LoWPAN을 사용하는 BLE 네트워크이다. 이 네트워크는 라즈베리 파이로 구현된 CoAP 서버와 6LoWPAN 게이트웨이 및 아두이노로 구성되어 있다. 아두이노는 Arduino Uno 제품을 사용했으며 온도 센서와 연결했다. 아두이노에서 실시간으로 센서가 보내주는 아날로그 측정값을 받아 디지털 온도로 변환하는 작업을 수행하고, 시리얼 통신 (Universal Asynchronous Receiver/Transmitter, UART)으로 CoAP 서버와 통신한다. 그리고 CoAP 서버는 아두이노로 전송받은 온도값을 자신의 CoAP 리소스 트리에 반영한다. 이후 CoAP 클라이언트가 요청할 시, 리소스 트리의 가장 최근 업데이트 된 값을 전송한다. 6LoWPAN 게이트웨이는 BLE 기반의 6LoWPAN 네트워크와 IPv6를 사용하는 인터넷망과 통신이 가능하도록 패킷을 변환 및 포워딩한다.

CoAP 서버 및 6LoWPAN 게이트웨이는 Raspbian OS가 설치된 라즈베리 파이를 이용하였다. 네트워크 B는 IPv6를 이용하는 이더넷 (Ethernet) 네트워크이며 CoAP 클라이언트가 존재한다. CoAP 클라이언트는 Windows OS를 사용하는 PC를 사용하였고, IPv6를 사용하는 인터넷망에 연결되어있다. CoAP 클라이언트, 서버 두 장치 모두 JAVA 기반의 오픈소스인 Californium CoAP Framework를 사용하였다.

Code	Description
# modprobe bluetooth_6lowpan	Enable 6LoWPAN module in Linux
# mount -t debugfs none /sys/kernel/debug	Enable kernel debugging
# echo 35 > /sys/kernel/debug/bluetooth/6lowpan_psm	Set Bluetooth PSM for IPSP profile
# echo "connect 00:11:22:33:44 1" > /sys/kernel/debug/bluetooth/6lowpan_control	Create 6LoWPAN connection between two node
# ifconfig eth0 inet6 add 2001:db8:abdc:1::1/64 # ifconfig bt0 inet6 add 2001:db8:abdc:2::1/64	Assign IPv6 address to 6LoWPAN gateway
# ifconfig bt0 inet6 add 2001:db8:abdc:2::2/64	Assign IPv6 address to CoAP server
# echo 1 > /proc/sys/net/ipv6/conf/all/forwarding	Enable IPv6 routing of 6LoWPAN gateway
# route -6 add default gw 2001:db8:abdc:2::1 netmask 256 dev bt0	Add default gateway of CoAP server

그림 3. CoAP 서버와 게이트웨이 간 6LoWPAN 통신을 하기 위한 과정
 Fig. 3. The process to communicating 6LoWPAN gateway and CoAP Server

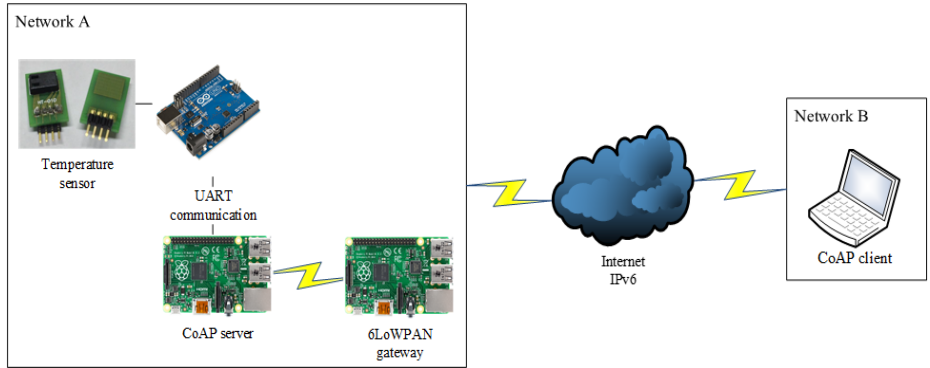


그림 4. 테스트베드 구조도
Fig. 4. Testbed architecture

2. 실험방법

본 논문에서는 3가지 실험을 수행하였다. 첫 번째 실험은 네트워크 A의 CoAP 서버와 6LoWPAN 게이트웨이 간 통신 가능 여부를 확인하고, 네트워크 B의 CoAP 클라이언트와 네트워크 A의 CoAP 서버 간 통신 가능 여부를 확인하였다. 두 번째 실험은 네트워크 B의 CoAP 클라이언트와 네트워크 A의 CoAP 서버 간의 CoAP 통신을 수행하였다. 마지막으로 세 번째 실험은 BLE 네트워크에 구현한 6LoWPAN 기반의 CoAP과 기존에 활용되고 있는 통신기술인 HTTP 간 성능 분석을 수행하였다.

첫 번째 실험은 네트워크 A의 CoAP 서버와 6LoWPAN 게이트웨이 간 Internet Control Message Protocol version 6 (ICMPv6) 패킷을 주고받으며 네트워크 구성이 잘 되었는지 확인하였다. 이후, 네트워크 A의 CoAP 서버와 네트워크 B의 CoAP 클라이언트 간 ICMPv6 패킷을 주고받으며 인터넷망을 통한 통신이 이루어짐을 확인하였다. 이때

6LoWPAN 게이트웨이에서 6LoWPAN 패킷과 인터넷 패킷이 어떻게 상호 변환되는지 확인하였다.

두 번째 실험은 CoAP 클라이언트가 CoAP 서버의 온도 센서의 측정값을 받아오는 시나리오를 구성했다. CoAP 서버는 아두이노와 UART 통신을 수행하며, 아두이노가 전송하는 최근 측정된 온도를 수신하고 자신의 리소스를 업데이트한다. 이 실험으로 CoAP 패킷이 인터넷망과 6LoWPAN 망에서 어떻게 전송이 되는지 확인할 수 있었다.

마지막으로 세 번째 실험은 네트워크 A의 CoAP 서버에 추가로 HTTP 서버를 구성한다. 그리고 CoAP 서버와 6LoWPAN 게이트웨이 간 HTTP와 CoAP 프로토콜의 성능 분석을 수행 하였다.

3. 실험결과 및 분석

그림 5는 네트워크 A에서 6LoWPAN 게이트웨이와 CoAP서버 간의 ICMPv6 패킷을 주고받은 결과이다. 64바

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::21a:7dff:feda:71...	fe80::21a:7dff:feda:7104	ICMPv6	120	Echo (ping) request id=0x0637, seq=1, hop limit=64 (reply in 2)
2	0.121436	fe80::21a:7dff:feda:71...	fe80::21a:7dff:feda:710c	ICMPv6	120	Echo (ping) reply id=0x0637, seq=1, hop limit=64 (request in 1)
3	1.001131	fe80::21a:7dff:feda:71...	fe80::21a:7dff:feda:7104	ICMPv6	120	Echo (ping) request id=0x0637, seq=2, hop limit=64 (reply in 4)
4	1.101765	fe80::21a:7dff:feda:71...	fe80::21a:7dff:feda:710c	ICMPv6	120	Echo (ping) reply id=0x0637, seq=2, hop limit=64 (request in 3)

```

> Frame 1: 120 bytes on wire (960 bits), 120 bytes captured (960 bits) on interface 0
> Linux cooked capture
> Internet Protocol Version 6, Src: fe80::21a:7dff:feda:710c, Dst: fe80::21a:7dff:feda:7104
  Internet Control Message Protocol v6
    Type: Echo (ping) request (128)
    Code: 0
    Checksum: 0x6b99 [correct]
    Identifier: 0x0637
    Sequence: 1
    [Response In: 2]
    Data (56 bytes)
    
```

그림 5. 6LoWPAN 게이트웨이와 CoAP 서버간의 ICMPv6 패킷 교환
Fig. 5. Packet transmission between 6LoWPAN gateway and CoAP Server

No.	Time	Source	Destination	Protocol	Length	Info
81	3.514387	2001:db8:abcd:2::2	2001:db8:abcd:1::2	ICMPv6	120	Echo (ping) request id=0x092c, seq=1, hop limit=64 (reply in 82)
66	3.514503	2001:db8:abcd:2::2	2001:db8:abcd:1::2	ICMPv6	118	Echo (ping) request id=0x092c, seq=1, hop limit=63 (reply in 67)
67	3.514976	2001:db8:abcd:1::2	2001:db8:abcd:2::2	ICMPv6	118	Echo (ping) reply id=0x092c, seq=1, hop limit=64 (request in 66)
82	3.515011	2001:db8:abcd:1::2	2001:db8:abcd:2::2	ICMPv6	120	Echo (ping) reply id=0x092c, seq=1, hop limit=63 (request in 81)

> Frame 81: 120 bytes on wire (960 bits), 120 bytes captured (960 bits) on interface 1

> Linux cooked capture

> Internet Protocol Version 6, Src: 2001:db8:abcd:2::2, Dst: 2001:db8:abcd:1::2

> Internet Control Message Protocol v6

> Ethernet II, Src: AsustekC_ae:d8:7a (1c:b7:2c:ae:d8:7a), Dst: Raspberr_75:48:b9 (b8:27:eb:75:48:b9)

> Internet Protocol Version 6, Src: 2001:db8:abcd:1::2, Dst: 2001:db8:abcd:2::2

> Internet Control Message Protocol v6

그림 6. 6LoWPAN 패킷을 인터넷망 패킷으로 변환된 결과
 Fig. 6. The packet transition result from 6LoWPAN to Internet

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000	2001:db8:abcd:1::2	2001:db8:abcd:2::2	CoAP	89	CON, MID:26706, GET, TKN:ac 6a e1 ce b3 8c, /temperature/room
2	0.105	2001:db8:abcd:2::2	2001:db8:abcd:1::2	CoAP	79	ACK, MID:26706, 2.05 Content, TKN:ac 6a e1 ce b3 8c (text/plain)

> Frame 1: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface 0

> Ethernet II, Src: AsrockIn_2c:c6:fd (d0:50:99:2c:c6:fd), Dst: Raspberr_75:48:b9 (b8:27:eb:75:48:b9)

> Internet Protocol Version 6, Src: 2001:db8:abcd:1::2, Dst: 2001:db8:abcd:2::2

> User Datagram Protocol, Src Port: 50268 (50268), Dst Port: 5683 (5683)

> Constrained Application Protocol, Confirmable, GET, MID:26706

01.. = Version: 1

..00 = Type: Confirmable (0)

.... 0110 = Token Length: 6

Code: GET (1)

Message ID: 26706

Token: ac6ae1ceb38c

> Opt Name: #1: Uri-Path: temperature

> Opt Name: #2: Uri-Path: room

그림 7. 온도 센서의 측정값을 요청하는 패킷
 Fig. 7. The request packet to get the value of temperature sensor

이트의 페이로드 (Payload)를 서로 송수신하고 있으며 통신이 원활하게 이루어짐을 알 수 있다.

그림 6은 네트워크 A의 CoAP 서버가 설치된 장치가 네트워크 B의 CoAP 클라이언트로 ICMPv6 메시지를 주고받는 패킷을 6LoWPAN 게이트웨이에서 캡처한 화면이다. CoAP 서버가 CoAP 클라이언트로 ICMPv6 패킷을 보내면 6LoWPAN 게이트웨이에서 이 패킷을 수신한다. 이때 6LoWPAN 게이트웨이는 라우팅 테이블을 참고하여 자신에게 연결된 인터넷망으로 나가는 패킷일 경우 그에 맞는 패킷으로 변환하고 포워딩한다. CoAP 클라이언트에서 이 패킷을 수신하면 응답 패킷을 CoAP 서버로 전송한다. 이때도 역시 6LoWPAN 게이트웨이에서 패킷을 수신하고 라우팅 테이블을 참고하여 자신이 담당하고 있는 네트워크로 전송되는 패킷이면 6LoWPAN 패킷으로 변환하여 포워딩한다.

그림 7은 CoAP 클라이언트에서 CoAP 서버로 온도 센서

에서 측정된 값을 요청하는 메시지를 보여준다. 해당 메시지는 CON 메시지를 활용하였으며, 그림에서 CoAP 헤더에 포함된 메시지 타입 (Type), 길이, 메소드 코드 등의 다양한 정보를 확인할 수 있다.

그림 8은 요청된 CON 메시지에 대한 응답 메시지를 보여준다. CoAP 서버가 클라이언트로부터 CON 메시지를 전송받았기 때문에 그에 대한 응답으로 ACK 메시지를 보낼 수 있다. 또한, piggy-backed response 방식을 활용하여 ACK 메시지에 요청에 대한 응답 메시지에 대한 정보인 온도 정보가 패킷에 포함되어 있음을 확인할 수 있다.

그림 9는 BLE 네트워크에 구현한 6LoWPAN 환경에서 0바이트의 페이로드 (Payload)를 가지는 메시지를 100회 전송할 때 HTTP와 CoAP의 평균 전송시간을 비교한 결과이다. 같은 응용을 구현했을 때 CoAP을 이용한 구현이 HTTP보다 약 21% 빠른 전송시간을 보이는 것을 알 수 있다.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000	2001:db8:abcd:1::2	2001:db8:abcd:2::2	CoAP	89	CON, MID:26706, GET, TKN:ac 6a e1 ce b3 8c, /temperature/room
2	0.105	2001:db8:abcd:2::2	2001:db8:abcd:1::2	CoAP	79	ACK, MID:26706, 2.05 Content, TKN:ac 6a e1 ce b3 8c (text/plain)


```

> Frame 2: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface 0
> Ethernet II, Src: Raspberr_75:48:b9 (b8:27:eb:75:48:b9), Dst: AsrockIn_2c:c6:fd (d0:50:99:2c:c6:fd)
> Internet Protocol Version 6, Src: 2001:db8:abcd:2::2, Dst: 2001:db8:abcd:1::2
> User Datagram Protocol, Src Port: 5683 (5683), Dst Port: 50268 (50268)
< Constrained Application Protocol, Acknowledgement, 2.05 Content, MID:26706
  01.. .... = Version: 1
  ..10 .... = Type: Acknowledgement (2)
  .... 0110 = Token Length: 6
  Code: 2.05 Content (69)
  Message ID: 26706
  Token: ac6ae1ceb38c
  > Opt Name: #1: Content-Format: text/plain; charset=utf-8
  End of options marker: 255
  > Payload: Payload Content-Format: text/plain; charset=utf-8, Length: 5
    
```

그림 8. 온도 센서의 측정값을 반환하는 패킷
 Fig. 8. The response packet contains the value of temperature sensor

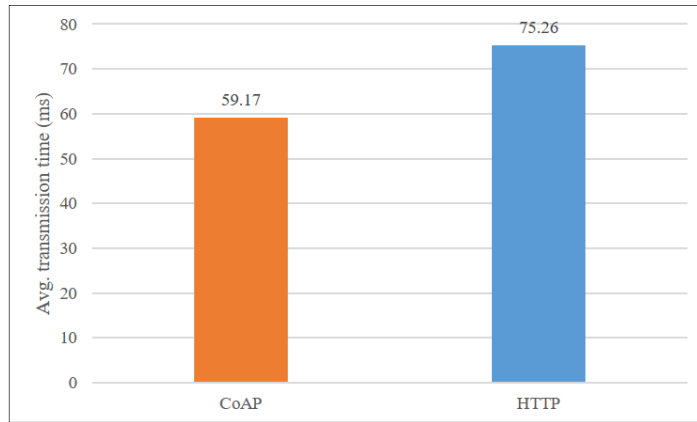


그림 9. 6LoWPAN 환경에서 CoAP과 HTTP의 평균 전송시간 비교
 Fig. 9. The comparison of average transmission time between CoAP and HTTP using 6LoWPAN

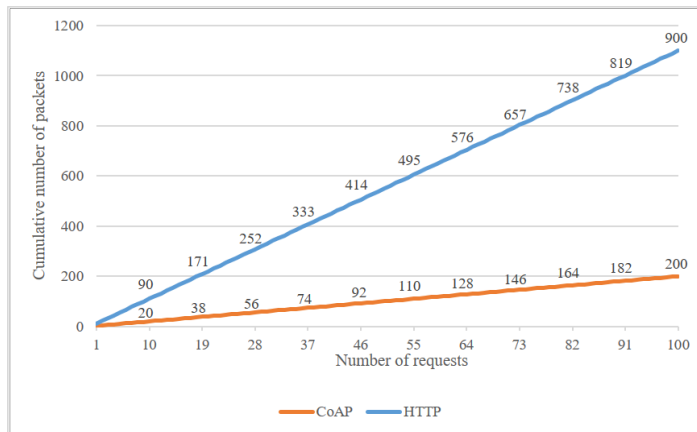


그림 10. 같은 요청 100번을 처리하는 데 사용된 패킷 수
 Fig. 10. The cumulative packet sequence number between CoAP and HTTP

그림 10은 같은 요청을 100번 처리하는 데 사용된 누적 패킷 수를 나타낸 결과이다. CoAP을 이용한 구현이 HTTP를 이용한 구현보다 약 22% 더 적은 패킷 수를 사용한다는 것을 보여준다. 이는 HTTP를 이용하여 통신할 때 TCP를 사용하므로, HTTP 통신 전후의 연결 수립과 연결 해제 작업에서 많은 오버헤드가 존재한다. 그에 비해 CoAP은 UDP를 사용하므로 TCP와 달리 CoAP 통신 이외에 필요한 작업이 없기 때문이다.

그림 11은 CoAP과 HTTP의 초당 처리 가능한 요청 수를 보여준다. CoAP이 HTTP보다 약 2.04배 많이 처리 가능한 것을 알 수 있으며 이는 CoAP이 더 적은 수의 패킷과 더 간결한 패킷 구조를 가지기 때문에 이와 같은 결과를 낼 수 있음을 알 수 있다.

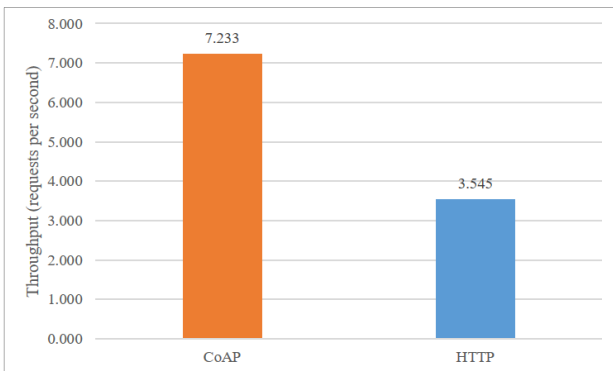


그림 11. CoAP과 HTTP의 요청 처리량 비교
Fig. 11. The comparison of throughput between CoAP and HTTP

V. 결론

본 논문에서는 BLE 네트워크 상에서 사물인터넷 서비스 제공을 위한 CoAP과 6LoWPAN을 활용한 통신기법을 제안하였다. 기존의 PAN 네트워크인 BLE 통신 기반 장치들은 링크 계층 기술을 사용하기 때문에, 사물인터넷처럼 인터넷을 이용한 서비스를 제공하기에 부적합하였지만, 본 논문에서 제안된 통신기술은 6LoWPAN을 활용한 IP 통신을 제공함으로써 이를 기반으로 한 다양한 서비스를 제공할 수 있다.

또한, 본 논문에서는 제안된 통신기술의 성능을 확인하기 위해 6LoWPAN 기술을 활용한 CoAP 프로토콜을 구현하여 기존의 HTTP를 이용한 통신과 성능을 비교·분석했다. 그 결과 CoAP이 HTTP보다 약 2배의 우수한 처리량을 보이며 평균 전송시간도 약 21% 빠름을 알 수 있었다. 뿐만 아니라 전송되는 패킷의 수도 CoAP이 HTTP보다 약 22% 적음을 알 수 있었다.

최근 방송통신 융합서비스가 주목을 받고 있다. 이러한 융합서비스 제공을 위해 DMB, 스마트폰 등의 방송수신 장치와 소형 센서 등의 사물인터넷 장치간에 통신이 활발히 이루어질 것으로 기대된다. 이에 본 논문에서 제시하는 사물인터넷 통신기능 구현 사례가 참조모델로 활용될 수 있으며, 사회 안전, 의료, 학습 등의 다양한 융합 서비스 창출에 기여를 할 수 있을 것이라 사료된다.

참고 문헌 (References)

- [1] W. Andrew and A. Anurag, The Internet of Things - A survey of topics and trends, Information Systems Frontiers, Volume 17, Issue 2, pp. 261-274, April, 2015
- [2] A. Aamir, S. Mohamed, S. Neelam and T. Anurag, An Internet of Things approach for motion detection using Raspberry pi, Intelligent Computing and Internet of Things (ICIT), 2014 International Conference on, pp 131-134, Jan. 2015.
- [3] W. Jiafu, L. Di, Z. Caifeng and Z. Keliang, M2M Communications for Smart City: An Event-Based Architecture, Computer and Information Technology (CIT), 2012 IEEE 12th International Conference on, pp. 895-900, Oct. 2012.
- [4] H. Stephan, S. Alexandru, B. Martin and C. Francois, A Domain Model for the Internet of Things, Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCOM), IEEE International Conference on and IEEE Cyber, Physical and Social Computing. pp. 411-417, Aug. 2013.
- [5] M. Geoff, The 6LoWPAN Architecture, EmNets '07 Proceedings of the 4th workshop on Embedded networked sensors, pp. 78-82, June, 2007
- [6] K. Seok-gap, P. Il-kyun, S. Seung-cheol and L. Byeong-tak, Trends of IETF CoAP Based Sensor Connection Protocol Technology, 2013 Electronics and Telecommunications Trends, ETRI, pp. 133-140, Dec. 2013
- [7] N. Kushalnagar, G. Montenegro and C. Schumacher, IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, And Goals, Internet Engineering Task Force (IETF) Request for Comments: 4919, Oct. 2015

- [8] Z. Shelby, K. Hartke and C. Bormann, The Constrained Application Protocol, Internet Engineering Task Force (IETF) Request for Comments: 7252, June, 2014
- [9] M. Kovatsch, S. Duquennoy and A. Dunkels, A Low-Power CoAP for Contiki, Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on, pp. 855 - 860, Oct. 2011
- [10] L. Alessandro, M. Pol and Anna Calveras, TinyCoAP: A Novel Constrained Application Protocol (CoAP) Implementation for Embedding RESTful Web Services in Wireless Sensor Networks Based on TinyOS, Journal of Sensor and Actuator Networks, pp. 288 - 315, 2013
- [11] T. Fielding, T. Richard, Principled Design of the Modern Web Architecture, ACM Transactions on Internet Technology (TOIT), Vol. 2, Issue 2, pp. 115-150, May, 2002
- [12] S. Teemu, K. kanji, B. Frank, D. Joe and H. Robin et al., Internet Protocol Support Profile, Bluetooth Specification, Dec. 2014

저 자 소 개



김 철 민

- 2015년 : 경북대학교 컴퓨터학부 공학사
- 2015년 ~ 현재 : 경북대학교 컴퓨터학부 석사과정
- ORCID : <http://orcid.org/0000-0002-5121-440X>
- 주관심분야 : 사물인터넷



강 형 우

- 2011년 : 경북대학교 컴퓨터학부 공학사
- 2013년 : 경북대학교 컴퓨터학부 공학석사
- 2013년 ~ 현재 : 경북대학교 컴퓨터학부 박사과정
- ORCID : <http://orcid.org/0000-0003-2430-4570>
- 주관심분야 : 사물인터넷, 이동성관리, 미래인터넷



최 상 일

- 2010년 : 경북대학교 공학사
- 2012년 : 경북대학교 컴퓨터학부 공학석사
- 2012년 ~ 현재 : 경북대학교 컴퓨터학부 박사과정
- ORCID : <http://orcid.org/0000-0003-3473-6131>
- 주관심분야 : 사물인터넷, 가시광통신



고 석 주

- 1992년 : KAIST 공학사
- 1994년 : KAIST 공학석사
- 1998년 : KAIST 공학박사
- 1998년 ~ 2004년 : ETRI 표준연구센터 선임연구원
- 2004년 ~ 현재 : 경북대학교 컴퓨터학부 교수
- ORCID : <http://orcid.org/0000-0003-3429-2040>
- 주관심분야 : 미래인터넷, 사물인터넷, IP Mobility, 멀티캐스트