

## 웹 서비스 기반 IoT 프록시 서버

김문성\* · 최진우\* · 전병찬\*\* · 이상정\*

\*순천향대학교

\*\*청운대학교

## IoT Proxy Server based on Web Service

Mun-Seong Kim\* · Jin-Woo Choi\* · Byoung-Chan Jeon\*\* · Sang-Jeong Lee\*

\*SoonCheonHyang University

\*\*ChungWoon University

E-mail : chyin370@naver.com · jinwoo4130@naver.com

jbc66@chungwoon.ac.kr · sjlee@sch.ac.kr

### 요 약

최근 사물인식에 관한 관심이 증가하면서 인터넷으로 모든 사물들을 연결하여 웹 서비스를 통해 관찰, 감독, 제어하는 사물인터넷 환경이 보편화 되고 있다. 따라서 웹 서비스를 통한 사물인터넷 환경으로의 침입에 대비한 보안이 중요해 지고 있다. 또한 사물인터넷 환경을 주로 이루는 장치들 대부분은 저성능의 디바이스인 경우가 대부분이고 이 장치들 간의 통신을 위한 새로운 프로토콜로 CoAP 프로토콜이 등장하였다. 본 논문에서는 사물인터넷 환경에서의 보안을 위한 리버스 프록시와 웹 서비스와 사물인터넷을 연결하는 HTTP-CoAP 프로토콜을 변환하는 프록시 서버 시스템을 설계 개발한다.

### 키워드

사물인터넷, CoAP 프로토콜, 리버스 프록시, 프로토콜 변환

## 1. 서 론

인터넷 속도의 발전에 따라 사물들을 연결하여 감독하고 제어하는 사물인터넷 환경과 더불어 웹 서비스의 중요성이 최근 들어 중요한 화두로 떠오르고 있다. 따라서 사물인터넷 환경에서의 보안 문제와 제한된 환경의 사물 사이의 새로운 통신기술이 나타

나고 있다. 또한 기존의 웹 서비스를 통하여 사용자가 사물을 모니터링하고 제어하는 사물 인터넷 환경이 보편화 되고 있다 [1].

웹 서비스를 통한 사물인터넷 환경에서 보안과 제한된 환경의 통신기술을 지원하기 위한 하나의 방법으로 프록시 서버를 이용해 사용자에게 보다 나은 서비스를 제공하는 것이 제시되고 있다 [2].

본 논문에서는 내부 사물인터넷 환경으로의 직접 접근을 차단하면서 제한된 자원을 갖는 사물 인터넷 환경과 외부 인터넷을 웹 기반으로 접속하는 프록시 서버를 설계 개발한다. 즉, 외부로 노출되는 웹 서비스와 내부에 구성되어 있는 사물인터넷을 분리하고, 제한적 네트워크 환경에서의 원활한 네트워크를 지원하는 CoAP 프로토콜과 기존의 HTTP 프로토콜과의 상호작용을 수행하여 웹 서비스에서 사물인터넷 환경으로의 진입을 제공하는 프록시 서버를 설계 및 구현한다.

## II. 사물인터넷 환경과 CoAP

### 2.1 사물인터넷 환경

사물인터넷, IoT (Internet of Things)는 컴퓨터뿐만 아니라 다양한 사물, 데이터 및 서비스 등 모든 것을 네트워크에 연결하여 상호 간의 정보를 교환할 수 있는 하나의 패러다임을 말하며, 사물과 사물간의 통신 그리고 사물과 사용자간의 통신을 모두 아우르는 용어이다. 즉, 사물인터넷은 실제 세계와 가상 세계의 모든 정보를 통합하는 기능을 제공하며, 표준 웹 브라우저를 통하여 실제 세계의 상태와 근접하게 추정할 수 있다 [1].

현실 세계의 정보를 웹 서비스를 통해 제공하는 웹 서비스 기반의 사물인터넷 시스템이 보편화 되고 있는 추세이지만 역으로 웹 서비스를 통한 사물인터넷 환경으로의 침입에 대한 보안은 미비한 수준이다. 그리고 이 사물들은 크게는 컴퓨터와 스마트폰과 같은 고성능의 장치들부터 작게는 하나의 센서나 마이크로 디바이스들이 모두 연결되는 거대한 네트워크이기에 저성능의 장치들과 고성능의 장치들 사이의 원활한 통

신을 위한 통신 프로토콜의 필요에 의해 사물인터넷 환경을 위한 새로운 프로토콜 기술이 출현하였고, 이 프로토콜과 기존의 인터넷을 구성하던 프로토콜 간의 상호 운용을 위한 방안이 필요하다.

### 2.2 CoAP

IETF CoRE(Constrained RESTful Environments) 워킹그룹에서 사물인터넷 환경을 구성하는 제한된 성능의 디바이스를 지원하기 위해 애플리케이션 프로토콜 CoAP(Constrained Application Protocol)을 표준화 하였다 [3-5]. CoAP는 사물인터넷 환경에서의 제한된 성능의 장치간의 통신을 위해 개발된 특수한 웹 전송 프로토콜이며, UDP를 기반으로 한다. 또한 REST 아키텍처를 기반으로 두고 있기 때문에, 기존의 HTTP 프로토콜로의 변환 및 연동이 용이하여 사용자에게 웹 서비스로 실제 세계의 정보를 전달하는 것이 가능하다.

기존의 HTTP와 CoAP 프로토콜의 통신을 위해 HTTP-CoAP Mapping 문서에서는 URI 매핑, HTTP 미디어 타입과 CoAP 콘텐츠 형식의 매핑, 캐싱 및 시간 제한 등에 관련된 내용을 다루고 있으며 [4], CoAP 기반 장치와 CoAP를 지원하지 않는 장치 간 통신을 위한 크로스-프로토콜을 통한 원활한 웹 서비스의 지원을 위한 용도로 프록시 사용을 제안하였다 [5].

## III. IoT 프록시 서버

### 3.1 프록시 서버

서버와 클라이언트 사이에서 자신을 통해서 다른 네트워크 서비스에 간접적으로 접속할 수 있게 해주는 컴퓨터나 응용 프로그램

램이며 클라이언트의 요청에 대해 대리 통신을 수행하는 기능을 프록시라고 하며, 그 중계 기능을 하는 서버 프로그램이 프록시 서버라 한다.

사물인터넷 환경에서 프록시는 익명의 사용자가 내부 사물인터넷 환경으로의 직접 접근을 차단한 간접 접근으로 보안을 강화하는 기능과 웹 서비스와 제한된 자원의 사물인터넷 환경을 연결하는 HTTP-CoAP 프로토콜 변환하는 기능을 제공한다. 일반적으로 프록시는 사용하는 위치에 따라 클라이언트 측을 위한 포워드 프록시(forward proxy)와 서버 측을 위한 리버스 프록시(reverse proxy)로 구분된다. 그림 1은 일반적인 프록시를 사용한 웹 서비스 시스템 구조를 보여주고 있다.

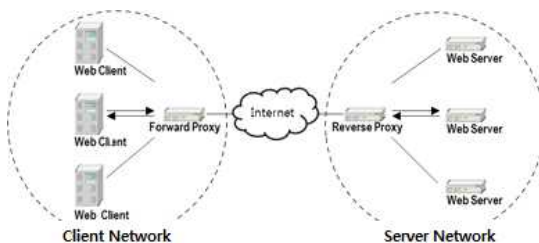


그림 1. 일반적인 프록시 서버를 사용한 웹 시스템 구조

그림 1에서 포워드 프록시는 클라이언트에 의해 설정되고 클라이언트가 외부 서버로 요청 시 외부 서버가 아닌 프록시 서버로 요청을 보내고, 요청받은 프록시 서버가 대신하여 외부 서버에 요청을 전달하고 응답을 받는다. HTTP-CoAP 프로토콜 변환을 지원하는 프록시 서버의 동작 방식은 포워드 프록시와 같다. 리버스 프록시는 서버 측의 프록시로 외부 클라이언트의 요청을 받아 서버로 해당요청을 전송하고, 서버의 응답을 다시 외부 클라이언트에게 응답하여 서비스를 제공한다. 또 클라이언트가 외부

서버로 요청을 보낼 때에는 프록시 서버의 자원 주소를 가지고 요청한다. 응용 서버의 자원들은 프록시 서버의 배후로 감추어지는 효과가 있기 때문에 사물인터넷 환경에서 내부 네트워크로의 직접 접근을 방지하여 보안성을 향상시킨다.

### 3.2 리버스 프록시 서버

본 논문에서는 사물인터넷 환경의 내부 네트워크를 숨기고 외부의 노출을 최소화하는 방식으로 보안 수준을 높일 수 있도록 리버스 프록시를 그림 2와 같이 구현 하였다.

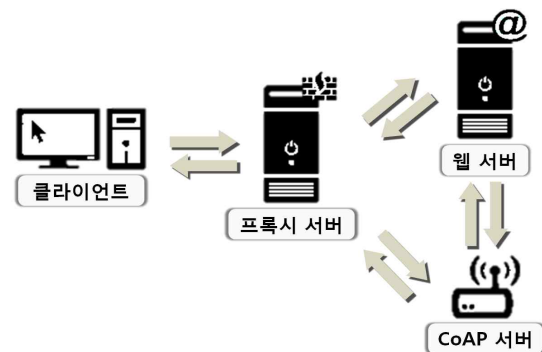


그림 2. IoT 환경에서의 웹 서비스를 위한 리버스 프록시 시스템

그림 2에서의 프록시 서버는 외부로 노출을 최소화하여 프록시의 후면에 위치해 있는 웹 서버와 CoAP 서버를 감추어 클라이언트에서의 직접 접근을 방지한다. 리버스 프록시의 구현 및 테스트를 위하여 그림 3과 같은 사물인터넷 환경을 개발 구현하였다. 그림 3에서 마이크로 프로세서 Arduino Uno와 먼지, CO<sub>2</sub>, 조도, 온·습도 센서를 이용하여 값을 측정 후 임베디드 서버 BeagleBone Black(Ver. Rev C)을 통해 데이터를 가공 및 전송하여 웹 서버에 저장된다. 웹 서버는 저장된 데이터를 이용하여 다양한 형태로 사용자에게 제공한다.

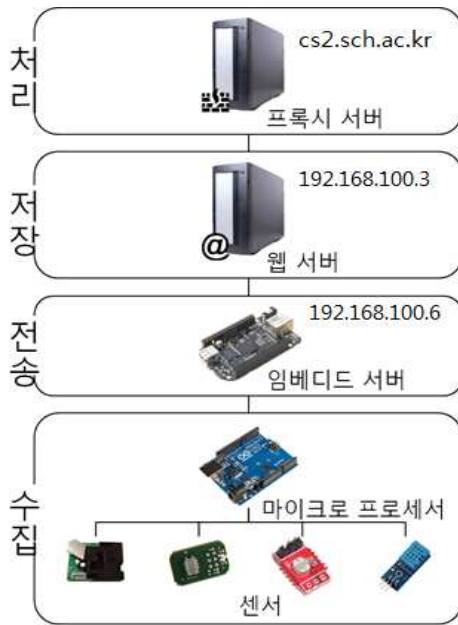


그림 3. 리버스 프록시의 구현 시스템 구성도

그림 3에서 웹 서버는 아파치 2.0, 프록시 서버는 아파치 HTTP 프록시 모듈을 사용하여 구현하였다. 그림 4는 리버스 프록시를 추가 설치한 시스템의 구현된 결과를 보여 주고 있다.



그림 4. 리버스 프록시의 구현 테스트

그림 4의 주소창에 표시된 것과 같이 프록시 서버의 URI를 통하여 웹 서버로의 요청이 가능하고, 내부의 웹 서비스를 하는

서버의 주소는 노출되지 않는다.

### 3.3 HTTP-CoAP 변환 프록시

사물인터넷 환경에서 노드들의 통신을 원활하게 하기 위한 CoAP 프로토콜과 웹 서비스를 제공하기 위한 HTTP 프로토콜을 상호 변환해주는 중계 역할로써 프록시 서버를 구현 하였다. 그림 5는 클라이언트가 CoAP 서버로의 요청을 하였을 때, 프록시의 동작과정을 도식화한 다이어그램이다.

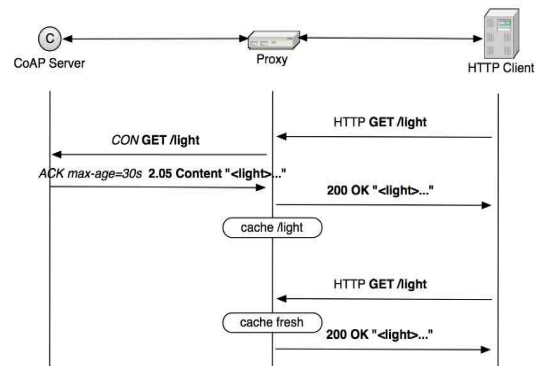


그림 5. HTTP-CoAP 프로토콜 변환 프록시의 동작

그림 5와 같이 HTTP-CoAP 변환 프록시를 통하여 사물인터넷 환경의 노드가 가지고 있는 현재의 가장 근접한 실제 세계의 센서 정보를 웹 서비스를 통해 제공할 수 있다.

그림 3에서 임베디드 서버가 HTTP-CoAP 변환 프록시 서버 기능을 수행한다. 임베디드 서버는 BeagleBone Black(Ver. Rev C)을 사용하여 보드 상에 우분투 리눅스를 탑재하고, C언어 기반의 Libcoap를 이용하여 서버를 구현하였다. 프록시 서버는 squid3 [7]와 HTTP-CoAP 매핑 모듈을 이용하여 구현하였다. squid3는 클라이언트의 요청을 서버로 전달하는 기본적인 프록시 기능부터, 트래픽 제어, 웹 캐싱, 프로토콜 변환 등을 지

원하는 오픈 소스 소프트웨어 프록시 서버이다. 그림 6은 구현된 테스트의 결과를 보여준다.

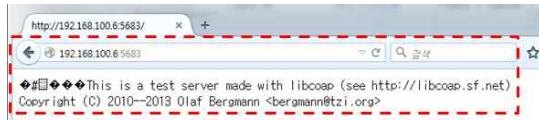


그림 6. HTTP-CoAP 변환 프록시 서버의 구현 테스트

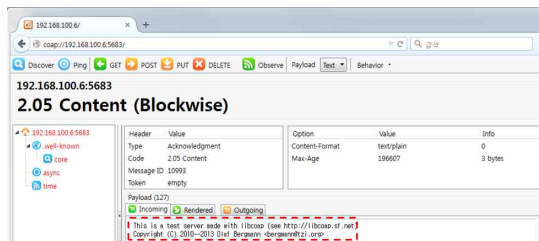


그림 7. CoAP 서버의 작동 테스트

그림 6의 프록시 서버 구현 테스트에 사용된 CoAP 서버는 Libcoap에 내장된 예제 서버 프로그램을 사용하였으며, 해당 서버 프로그램은 요청 시 기본 메시지를 응답하는데, 웹 브라우저에서 CoAP 프로토콜의 서비스 포트번호인 5683을 사용하여 정보를 요청하였을 때, CoAP 서버에서 기본 메시지로 응답을 하는 것을 확인할 수 있었다.

그림 7은 Firefox 웹 브라우저에서 동작하는 CoAP 프로그램 테스트 도구인 Copper를 사용하였을 때의 서버의 응답 메시지이다. CoAP 서버에 GET 요청을 하였을 때, 서버에서의 테스트 응답하는 것을 확인할 수 있다.

### III. 결 론

본 논문에서 제안한 웹 서비스를 위한 사물인터넷 환경에서의 프록시 서버를 구현하였다. 리버스 프록시를 통하여 사물인터넷

의 내부 네트워크를 감추어 보안성을 향상시켰다. 또한 내부 네트워크에서의 CoAP 프로토콜을 통하여 저전력, 저성능의 장치들에서의 통신과 이기종의 HTTP 프로토콜을 사용하는 장치들과의 통신을 가능하도록 하였다.

다양한 사물인터넷 환경에서 본 논문에서 제안된 IoT 프록시를 사용하여 외부에 노출을 최소화하면서 내부로의 HTTP-CoAP 프로토콜 변환 프록시를 적용한다면 보안성의 향상과 함께 원활한 웹 서비스의 제공뿐만 아니라 저성능의 장치들로 구성되는 사물인터넷 환경으로 인해 제작비의 절감효과까지도 얻을 수 있을 것이다.

### 참고문헌

- [1] Lina Yao, Quan Z. Sheng, Schahram Dustdar, "Web-Based Management of the Internet of Things", IEEE Computer Society, p.10-17 Jan./Feb. 2015
- [2] 박지예, 강남희, "리버스 프록시 기반 IoT 서비스 도메인 설계", 한국인터넷 방송통신학회논문지, 제14권 제6호, p.1-6, 2014년 12월
- [3] Z. Shelby, K. Hartke, C. Bormann, "The Constrained Application Protocol (CoAP)", IETF RFC 7252.
- [4] A. Castellani, S. Loreto, A. Rahman, T. Fossati, E. Dijk, "Guidelines for HTTP-CoAP Mapping Implementations", IETF draft-ietf-core-http-mapping-07, April 2016.
- [5] 고석갑, 박일균, 손승철, 이병탁, "IETF CoAP 기반 센서 접속 프로토콜 기술 동향", 한국전자통신연구원, 전자통신동향분석, 2013년 12월
- [6] 고석갑, 오승훈, 손승철, 이병탁, "사물인터넷 표준 CoAP 기술 및 구현 동향", 정보통신기술진흥센터 주간기술동향, 2014년 6월
- [7] squid3, <http://www.squid-cache.org/>