

IoT Platform Implementation and Service Operating Method for Private Networks

Lee Kyoung Hun[†] · Kim DooYoung^{**} · Song Seheon^{***} · Lee Sangil^{****} · Park JaeHyun^{****}

ABSTRACT

Today various IoT platforms exist, but most of them only consider IPv6, without other types of network including private network. They support, therefore, the private network to use virtual private network or through the hardware gateway. To solve this limitation, we propose a IoT platform that provides IoT Services on private networks. Also we verify this platform though constructing a testbed.

Keywords : Internet of Things, Proxy, IoT Platform, Smart Building, Private Network

용이한 사설망 지원을 위한 IoT 플랫폼 구현 및 서비스 운영 방법

이 경 훈[†] · 김 두 영^{**} · 송 세 현^{***} · 이 상 일^{****} · 박 재 현^{****}

요 약

현재 출시된 다양한 사물인터넷(IoT) 플랫폼은 대부분 IPv6의 네트워크 환경을 고려하여 개발되었지만, 아직까지 IPv4에서 IPv6로의 전환율은 저조하다. 따라서 IPv4기반의 네트워크에서 운용되는 IoT 플랫폼은 IP부족현상을 해결하기 위해 기구축된 사설망을 지원해야 한다. 기존 IoT 플랫폼들은 전문가에 의해 가상 사설망을 이용하거나 하드웨어 스위치/AP(Access Point)를 이용하여 사설망을 지원하고 있다. 본 연구에서는 기존 IoT 플랫폼들이 사설망 환경을 지원하기 위해 필요한 전문가 활용 또는 하드웨어 도입에 따른 추가 비용 문제를 해결하고, 비전문가도 사설망 환경에서 IoT 서비스를 제공할 수 있는 IoT 플랫폼을 개발하였다. 또한 IoT 플랫폼을 테스트베드에 적용하여 IoT 플랫폼의 기능 검증을 수행하였다.

키워드 : 사물인터넷, 프록시, IoT 플랫폼, 스마트빌딩, 사설망

1. 서 론

사물인터넷(IoT)은 사물, 사람, 공간을 인터넷으로 연결하여, 정보의 생성·수집·공유·활용을 위한 사물·인간·서비스의 연결망이다[1]. IoT는 주요 3대 기술(센싱 기술, 유무선 통신 및 네트워크 인프라 기술, IoT 서비스 인터페이스 기술)로 구성되며 IoT 플랫폼은 센싱 기술과 유무선 통신 및 네트워크 인프라 기술에 의해 구현된 사물의 기능을 서비스로 제

공하기 위한 IoT 서비스 인터페이스 기술이다. IoT 플랫폼은 사물(Device) 연결부터 서비스 제공까지 폭넓은 기능을 제공해야 하며, 이러한 기능들을 제공하기 위해서 IoT 플랫폼은 일반적으로 서비스 플랫폼과 IoT 게이트웨이, IoT 디바이스로 구성된다[2].

현재 출시된 다양한 IoT 플랫폼들은 각 사물을 인터넷으로 접근하기 위해 IP기반 주소를 부여하고 있으며, IoT 환경에 존재하는 많은 개체를 수용하기 위해 IPv6를 고려하여 구현되었다[3]. 하지만 현재 국내의 IPv6 사용률(IPv6를 사용하여 Google에 접속한 통계)은 2%대에 달하며, 전 세계적으로도 10%대에 미치는 등 IPv6의 사용률은 저조한 상황이다[4]. 따라서 현재 대다수가 이용 중인 IPv4는 모든 사물에 IP를 부여하기엔 턱없이 부족한 상황이다. 이러한 문제점을 해결하기 위한 방법으로 NAT(Network Address Translation)와 DHCP(Dynamic Host Configuration Protocol)가 이용된다.

※ 이 논문은 민군기술협력사업(UMI3018RD1)으로 지원받았음.
† 정 회 원 : 메타빌드(주) 전임연구원
** 비 회 원 : 부울(주) 수석연구원
*** 정 회 원 : 메타빌드(주) 책임연구원
**** 비 회 원 : 국방과학연구소 책임연구원
Manuscript Received : January 29, 2016
First Revision : March 24, 2016
Second Revision : April 15, 2016
Accepted : April 15, 2016
* Corresponding Author : Park JaeHyun(foreshand@add.re.kr)

하지만, NAT는 사설 IP를 이용함으로써 IP 부족현상을 해결할 수 있으나 사설 IP를 이용하는 단말은 인터넷을 통해 접근이 불가능하다는 단점이 존재한다. 이와 같은 단점 때문에 NAT를 이용하는 네트워크 환경에서 IoT 서비스를 제공하기 위해서는 인터넷에서 사설 IP를 이용하는 단말에 접근 가능하도록 지원하는 하드웨어 스위치/AP(사물 연결 플랫폼, IoT 게이트웨이) 또는 전문가에 의해 구축된 가상 사설망(VPN: Virtual Private Network)을 이용해야만 한다. IP 부족현상을 해결하기 위한 또 다른 방법인 DHCP는 일정시간 IP를 대여하여 사용하는 방법으로 일정시간 경과 후 IP가 변경될 수 있어 IoT 환경의 사물의 IP가 변경되어 서비스가 불가능할 수 있는 문제를 내포하고 있으나 DHCP 옵션을 통해 간단히 문제를 해결할 수 있다.

본 연구에서는 IPv6의 저조한 사용률로 인해 발생하는 IPv4 네트워크(사설망)에서 IoT 환경 구축의 어려움을 해결하기 위해 1) 기존 방법의 가격 상승 요인을 줄이고, 2) 비전문가도 쉽게 IoT 환경을 구축할 수 있는 방법을 제안한다.

이러한 연구 목표를 달성하기 위해 사설망에서 IoT 서비스를 제공할 수 있는 소프트웨어 기반의 IoT 프록시를 연구 개발하였으며, IoT 프록시를 포함하는 새로운 개념의 IoT 플랫폼을 제안한다. 새로운 개념의 IoT 플랫폼의 구성요소는 다음과 같다.

- IoT 플랫폼: IoT 서비스를 제공하기 위한 플랫폼으로 본 연구에서는 IoT 서버, IoT 프록시, IoT 게이트웨이로 구성
- IoT 서비스: IoT 플랫폼에 의해 제공되는 서비스로 사물의 기능을 제 3의 사물·사람·서비스가 사용할 수 있도록 서비스화한 사물의 기능
- IoT 서버: IoT 서비스를 제공하기 위한 서버 플랫폼으로 외부 제 3 사물에 의해 서비스 요청 시 해당 서비스를 제공하는 IoT 프록시(게이트웨이)로 서비스 연계 기능과 IoT 게이트웨이 모니터링 기능, 서비스·IoT 게이트웨이 등의 상태 파악 및 새로운 사물을 추가할 수 있는 서비스 컴포넌트를 배포하는 기능 등을 제공
- IoT 프록시: 외부에서 접근이 불가능한 사설망 내에 존재하는 다수의 IoT 게이트웨이가 정상적으로 IoT 서비스를 제공할 수 있도록 프록시 기능을 제공하며 IoT 프록시는 외부의 서비스 요청을 수신한 후 해당 IoT 게이트웨이로 서비스 요청을 전달
- IoT 게이트웨이: IoT 디바이스를 인터넷과 연결하고 IoT 서비스를 실제로 제공하는 기능을 제공

본 연구에서는 위 정의된 IoT 플랫폼을 이용하여 사설망 환경에서 IoT 서비스를 제공하는 방법을 연구하고 시스템을 구현하였으며 이를 이용하여 스마트빌딩에 적용해봄으로써 기능 검증과 새로운 문제점들을 파악하였다.

이를 위해 본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 소개하고 기존 IoT 플랫폼의 한계점을 설명하며, 3장에서는 사설망의 특징을 소개하고 사설망 환경에서 IoT 환경 구축 한계점을 설명한다. 4장에서는 본 연구를 통해 구현된 사설망 환경을 지원하는 IoT 플랫폼을 소개한다. 5장에서는

IoT 플랫폼을 스마트빌딩에 적용하여 테스트한 결과에 대해 설명하고, 마지막으로 6장에서는 본 논문의 결론을 논한다.

2. 관련 연구

IoT가 대두되면서 국내외적으로 다양한 IoT 플랫폼이 존재하며 다양한 연구와 표준이 수행되었다. 국제 표준인 oneM2M[5]을 구현한 대표적인 IoT 플랫폼은 OCEAN[6]의 OpenMobius와 &Cube이며, 디바이스의 인터넷 연결에 중점을 둔 IoT 프레임워크로는 AllJoyn[7]이나 IoTivity[8]가 대표적이다.

사설망에 존재하는 단말에 접근하기 위한 관련 연구는 활발히 진행되어 왔다[9-11]. 하지만 기존 연구들은 IoT를 위한 연구가 아닌 범용적인 연구로 IoT 환경에서 효율이 떨어지거나[9, 10], 새로운 H/W를 추가하거나 가상 사설망(VPN)을 이용하는 연구가 진행되었다[11].

현재 다양한 IoT 플랫폼들[5-8]은 모든 사물을 인식할 수 있는 IPv6를 기반으로 구현되었다. IPv6를 기반으로 한 IoT 플랫폼들은 IPv4의 사설망을 지원하기 위해서 새로운 개념의 하드웨어(또는 소프트웨어) 스위치/AP(사물 연결 플랫폼, IoT 게이트웨이)를 추가하거나 전문가에 의해 가상 사설망을 이용한다. 본 연구에서는 이와 다르게 SW 기반의 IoT 프록시를 통해 사설망을 지원하도록 한다.

3. 사설망 환경의 특징과 IoT 환경 구축 한계점

오늘날의 네트워크 구성은 인터넷뿐 아니라 다양한 이유에 의해서 사설망(Private Networks, Intranet)을 구축하고 있다. 사설망은 IPv4의 IP 부족 현상을 해결하기 위해 고안된 기술들을 이용하여 구축된다. 사설망은 IP 부족 현상 해결 외에도 인터넷에서 직접적으로 사설망 내부 단말에 접근이 불가능하다는 특징 때문에 보안적인 측면으로도 사용된다.

IoT 환경에서 사설망은 사설망을 통해 인터넷에 연결된 사물에 접근이 불가능하므로 IoT 환경 구축에 어려움이 따른다. 사설망에서의 IoT 환경 구축에 따른 어려움을 해결하기 위해서 기존 IoT 플랫폼들은 새로운 새로운 개념의 하드웨어 스위치/AP를 추가하거나 전문가에 의해 가상 사설망을 이용하였다. 하지만 가정, 소규모 기업 등과 같은 네트워크 환경에서는 하드웨어 추가 또는 네트워크 전문가 이용이 부담스러울 수밖에 없다. 따라서 비전문가도 IoT 환경을 쉽게 구축할 수 있는 방법이 필요하다.

다음은 사설망을 구축하기 위한 기술들로 사설망 환경의 특징을 좀 더 자세하게 나타내고 있다.

3.1 NAT(Network Address Translation)

NAT는 IPv4의 IP 부족 문제를 해결하기 위한 하나의 방안이다. NAT는 네트워크를 구성하고 구성된 노드에 사설 IP(같은 네트워크에서만 사용 가능한 IP)를 부여하고 사설

IP를 이용해 인터넷에 접근하려하는 경우 사설 IP를 공인 IP(인터넷에서 사용 가능한 IP)를 부여하는 역할을 수행한다. NAT를 이용하는 경우 외부(인터넷 또는 다른 네트워크)에서는 내부에 노드를 확인할 수 없다. 즉, 인터넷에 연결된 사용자는 NAT 내부에 구성된 단말에 직접 접근이 불가능하다[12, 13].

3.2 DHCP(Dynamic Host Configuration Protocol)

인터넷 프로토콜 주소인 IP Address(이하 IP)는 여러 의미에 의해서 관리되고 있다. 이러한 관리 방법 중 하나인 DHCP는 관리자(DHCP 서버)가 IP를 요청한 단말에 IP를 대여해주는 방법이다. DHCP는 특별한 설정 없이 인터넷을 사용할 수 있기에 오늘날 많은 곳에서 DHCP를 사용하고 있다. 하지만 DHCP의 경우 동일한 단말이 동일한 IP를 할당받는 것을 보장하지 않는다(IP를 사용 중이라면 동일한 IP를 재할당 받을 수 있으나, DHCP 서버의 설정이나 IP 반환 등의 이유로 동일한 IP를 할당하는 것은 보장되지 않음). IoT 환경에서 사물(또는 도메인)의 IP가 바뀌는 경우 서비스 제공이 제한되어진다. 이러한 문제를 해결할 수 있는 IoT 도메인 관리자(디바이스)가 필요하다[14].

4. 사설망 환경을 지원하는 IoT 서비스 플랫폼

4.1 IoT 플랫폼 구현

본 시스템은 Fig. 1과 같이 IoT 서버, IoT 프록시, IoT 게이트웨이로 구성된다.

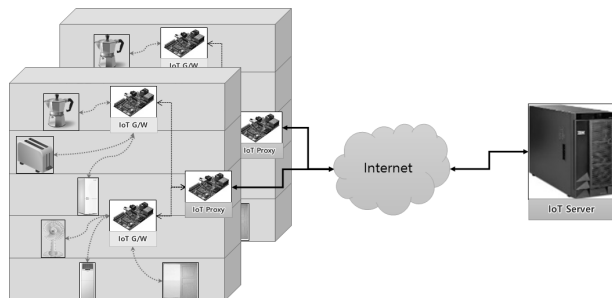


Fig. 1. A conceptual diagram of a IoT platform

1) IoT 게이트웨이

IoT 게이트웨이는 사물(Device)을 인터넷에 연결하고 각 사물의 기능을 서비스로 제공한다.

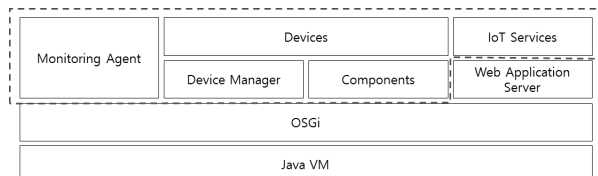


Fig. 2. IoT gateway architecture

본 연구에서는 Fig. 2와 같은 구조를 갖는 IoT 게이트웨이를 개발하였다. IoT 게이트웨이는 다양한 통신 방식을 갖는 사물을 인터넷에 연결할 수 있도록 다양한 통신 컴포넌트(Components)를 제공한다. 디바이스 관리자(Device Manager)는 통신 컴포넌트를 이용하여 연결된 사물을 객체화(Devices)하고 관리한다. IoT 게이트웨이는 웹서비스(SOAP) 형태로 사물 제어 기능을 제공하여, IoT 서버를 이용하지 않아도 IoT 서비스가 가능하며 통일된 제어 서비스를 제공하고, 3rd party IoT 서버를 지원이 가능하게 한다. 이를 위해 웹 서비스를 제공할 수 있는 WAS(Web Application Server)와 사물의 제어를 제공하는 서비스(IoT Services)를 추가하였다. 모니터링 에이전트(Monitoring Agent)를 통해 서버에 모니터링 정보를 제공하여 IoT 서버를 통해 IoT 게이트웨이의 상태, 서비스 상태 등을 모니터링할 수 있다. IoT 게이트웨이는 다양한 통신 컴포넌트와 객체화된 사물을 관리/제어하기 위해 OSGi를 이용하였다.

IoT 서비스는 앞서 언급한 바와 같이 웹서비스로 http://<게이트웨이 IP>:<고유 포트>/<IoT 서비스 경로> 형태의 URI를 이용해 제공되며 Fig. 4를 통해 IoT 서비스가 제공되는 형태에 대해서 상세히 확인이 가능하다.

2) IoT 프록시

IoT 프록시는 서비스를 제공하는 다수의 IoT 게이트웨이가 모두 인터넷에 노출될 수 없는 환경(IoT 게이트웨이가 사설망 내에서 서비스를 제공하는 환경)에서 IoT 프록시를 인터넷에 노출하여 서비스를 제공할 수 있도록 한다. IoT 프록시는 포트포워딩 기술을 이용하여 인터넷에 노출되며, 이를 통해 동일한 포트를 다수의 다른 노드로 포트포워딩할 수 없는 한계점을 해결하고 사설망 환경에서 IoT 서비스를 제공할 수 있도록 한다.

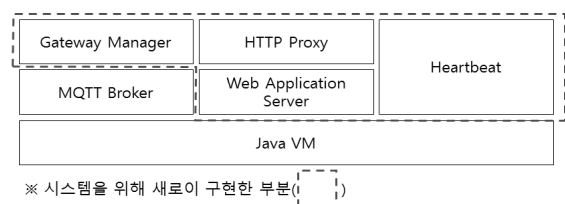


Fig. 3. IoT proxy architecture

IoT 프록시는 Fig. 3과 같은 구조로 IoT 플랫폼에서 IoT 게이트웨이와 IoT 서버의 중계기 역할을 수행한다. IoT 프록시는 게이트웨이 관리자(Gateway Manager)를 이용하여 MQTT(MQ Telemetry Transport) Topic으로 전달된 IoT 게이트웨이의 정보와 사물 데이터를 IoT 서버로 전달하며, IoT 서비스 요청은 HTTP Proxy를 이용하여 수신한다. HTTP Proxy는 저장하고 있는 IoT 게이트웨이 ID와 IP Address 정보를 이용하여 요청된 IoT 서비스를 해당 IoT 게이트웨이로 전달한다. IoT 프록시는 HTTP Proxy를 이용하여 수신한 IoT 서비스 요청의 URI를 변환하여 IoT 게이트웨이로 요청을 전달한다. IoT 게이트웨이로 요청을 전달하는 방법은 Fig. 4와 같이

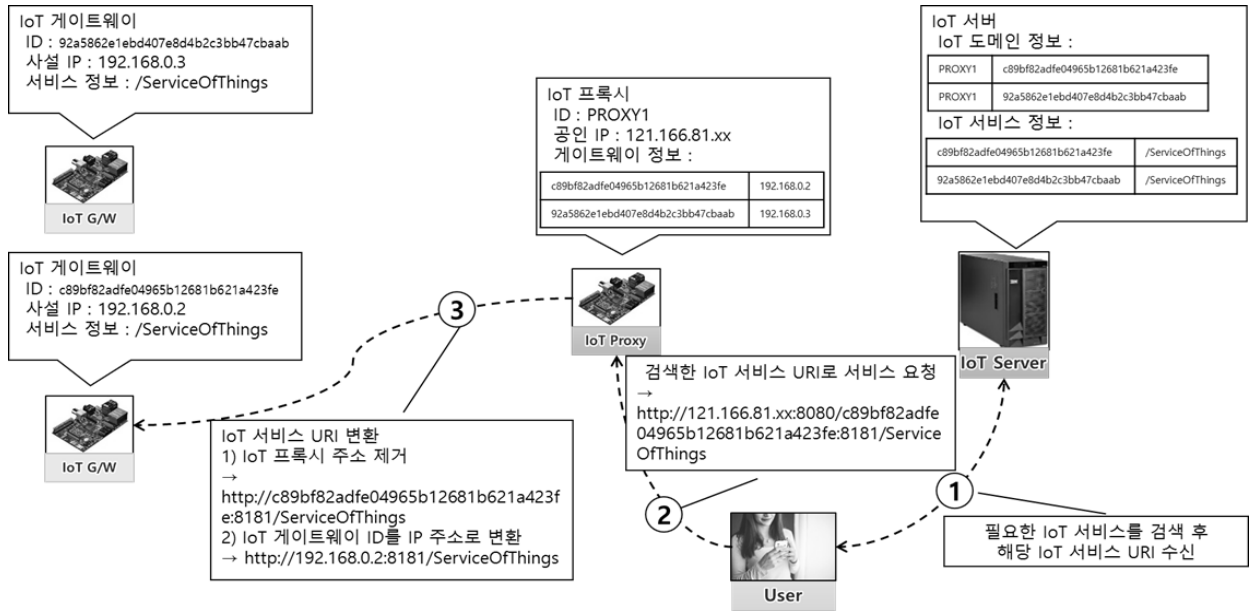


Fig. 4. The working process of the IoT proxy

수신한 URI를 IoT 게이트웨이에서 제공하는 서비스 URI로 변경하여 IoT 서비스를 호출하는 방식을 이용한다. IoT 프록시는 DHCP 환경의 한계를 해결하기 위해 주기적으로 자신의 상태 정보(Heartbeat)를 IoT 서버로 전송하여 자신이 속한 사설망의 공인 IP를 IoT 서버에서 인식할 수 있도록 한다. IoT 프록시를 통해 구분된 IoT 도메인은 IoT 게이트웨이의 도메인(위치)을 구분하고, 도메인 구분을 통해 사용자마다 각자의 IoT 서비스만을 제공할 수 있다. 예로 Fig. 1을 보면 IoT 게이트웨이는 도메인을 구분하고 있는 IoT 프록시에 연결되어 각각 다른 IoT 도메인을 구성한다.

3) IoT 서버

IoT 서버는 IoT 서비스와 IoT 게이트웨이를 보다 효율적으로 이용, 관리할 수 있는 기능을 제공한다. Fig. 5와 같이 IoT 서버는 IoT 프록시에서 전송되는 상태정보를 수신하기 위한 서버(Heartbeat Server)와 비동기 데이터를 처리하기 위한 메시지 브로커(JMS Broker), IoT 게이트웨이의 컴포넌트를 효율적으로 관리하기 위한 Repository로 구성된다. 또 IoT 서버는 IoT 서비스를 효율적으로 이용, 관리할 수 있도록 모니터링 웹 콘솔과 IoT 서비스를 사용할 수 있는 IoT 서비스 웹 앱(IoT Service Web App)을 제공한다.

IoT 서버는 IoT 서비스와 IoT 게이트웨이를 효율적으로 관리하기 위하여 IoT 프록시를 기반으로 생성된 IoT 도메인을 이용하여 IoT 서비스와 IoT 게이트웨이를 관리한다.

IoT 게이트웨이로부터 전달된 모니터링 정보는 모니터링(Monitoring) 기능을 통해 관리자와 사용자에게 제공된다. 모니터링 기능은 Fig. 8(left)와 같이 IoT 도메인으로 모니터링 정보를 분리하여 IoT 사용자가 해당 IoT 도메인의 상태(IoT 게이트웨이 리소스/네트워크 상태 정보, IoT 서비스 상태 정보 등)를 실시간으로 확인이 가능하도록 하며 간단

한 IoT 서비스를 이용할 수 있도록 구성하였다. 또 사용자가 위치와 시간과 상관없이 자신의 IoT 도메인의 IoT 서비스를 실행할 수 있도록 Fig. 8(right)와 같이 웹앱을 제공한다.

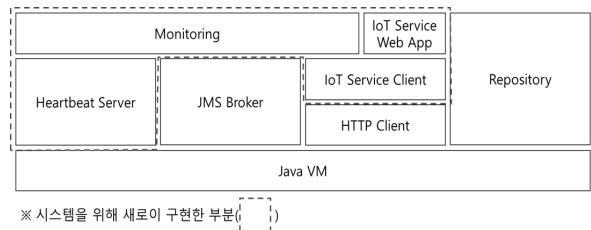


Fig. 5. IoT server architecture

4.2 IoT 플랫폼을 활용한 사설망 환경에서의 IoT 서비스 운용

본 시스템의 구성과 각각의 역할 및 기능은 앞서 설명하였다. 이번 절에서는 IoT 플랫폼을 이용하여 사설망에서 IoT 서비스를 운용하는 자세한 방식에 대해서 설명하려고 한다. IoT 플랫폼을 이용하여 IoT 서비스를 운용하는 과정은 IoT 게이트웨이 및 IoT 프록시가 구동되어 IoT 서비스가 등록되는 과정과 IoT 서버를 이용한 IoT 서비스 실행 과정으로 나누어 설명한다.

1) IoT 서비스 등록

IoT 서버는 외부에서 검색 및 접속이 가능하도록 공개되어 있으며 IoT 프록시가 구동되면 IoT 프록시는 주기적으로 자신의 정보를 IoT 서버로 전달한다. IoT 게이트웨이는 구동과 동시에 주기적으로 자신의 정보를 IoT 프록시로 전달하며, IoT 프록시는 IoT 게이트웨이의 정보를 IoT 서버로 전달함으로써 IoT 서버에 IoT 게이트웨이를 등록한다. IoT 게이트웨이가 등록된 후에 IoT 게이트웨이는 IoT 서비스를 등록한다.

2) IoT 서비스 실행

IoT 서비스 호출자는 IoT 서비스 정보와 IoT 게이트웨이 이름 등을 통하여 IoT 서비스를 구분하여 호출해야만 한다. 즉, IoT 서비스 호출자는 호출하고자 하는 IoT 서비스가 속한 IoT 프록시와 IoT 게이트웨이 주소(IP) 정보를 알 수 없다. 따라서 IoT 서비스 호출자는 IoT 서버에서 호출하고자 하는 IoT 서비스를 검색하고 해당 IoT 서비스를 IoT 서버를 통하여 호출할 수 있다. IoT 서버는 IoT 호출자에 의해 호출된 IoT 서비스의 정보를 이용하여 실제 IoT 서비스를 제공하는 IoT 게이트웨이와 IoT 프록시를 검색하여 해당 호출 정보를 IoT 프록시로 전달한다. IoT 서비스 호출을 수신한 IoT 프록시는 호출된 IoT 서비스 정보를 바탕으로 실제 IoT 서비스를 제공하는 IoT 게이트웨이에 해당 호출 정보를 전달한다. IoT 게이트웨이는 IoT 서비스가 정의한 사물의 기능을 실행 후 그 결과를 IoT 프록시로 반환한다. IoT 프록시는 IoT 서버로 결과를 반환하며 IoT 서버는 IoT 서비스 호출자에게 IoT 서비스 호출 결과를 반환한다.

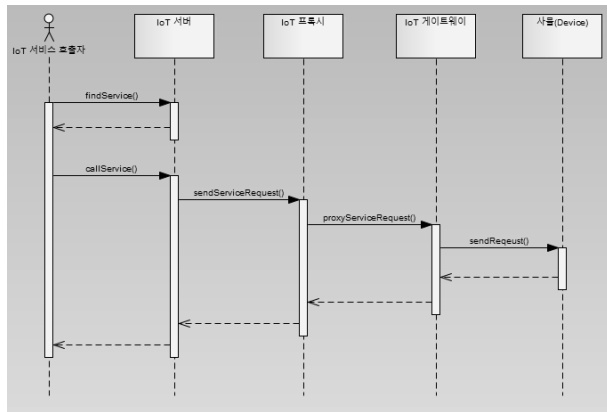


Fig. 6. A process of a IoT service request

IoT 서비스 호출자가 IoT 서버에서 서비스를 검색하여 원하는 서비스를 호출할 때에 URI는 `http://<프록시 IP>/<게이트웨이 ID>:<고유 포트>/<IoT 서비스 경로>`로 실제 IoT 게이트웨이에서 제공하는 서비스에 접근하는 URI와 차이가 존재한다. 이는 Fig. 4에서와 같이 IoT 프록시에 의해 IoT 게이트웨이가 제공하는 서비스에 접근 가능한 URI 주소로 변경되어 실제 IoT 서비스를 호출하게 된다.

5. 테스트베드를 이용한 IoT 플랫폼 검증

본 연구에서 개발된 IoT 플랫폼의 사설망 지원 및 IoT 서비스 제공 기능을 테스트하고 검증을 목표로 테스트베드를 선정하였으며 그 결과 IoT 플랫폼을 이용한 스마트빌딩을 적합한 테스트베드로 선정하였다. 스마트빌딩의 네트워크 구조는 일반적으로 빌딩 내부 네트워크(사설망)와 인터넷으로 구분되므로 본 연구에서 개발된 IoT 플랫폼의 기능 테스트 및 검증하기에 적합한 환경을 제공한다.

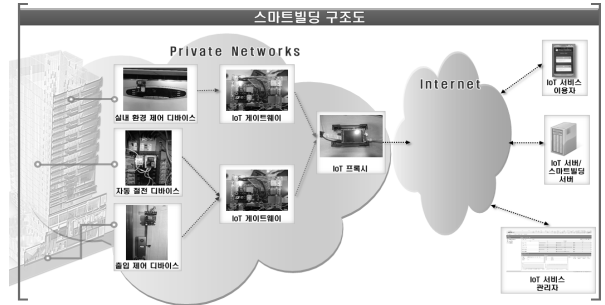


Fig. 7. The smart building block diagram

기존의 빌딩을 Fig. 7의 구조도와 같이 IoT 플랫폼을 적용하여 스마트빌딩을 구축하였다. 기존의 빌딩을 IoT 플랫폼을 이용하여 스마트빌딩으로 구축하기 위해서는 빌딩에 존재하는 사물(자원)을 제어하고 데이터 수집할 수 있는 새로운 사물 제어용 장치가 필요하였으며 이를 위해 오픈 하드웨어를 활용하여 사물 제어용 장치를 개발하여 이용하였다. 개발된 사물 제어용 장치와 사용된 오픈 하드웨어는 Table 1과 같다.

Table 1. The list of devices in the testbed

이름	사진	하드웨어
IoT 서버		PC
스마트빌딩 서버		
IoT 프록시		라즈베리파이
IoT 게이트웨이		라즈베리파이
출입문 제어 장치		Adafruit HUZAZH ESP8266 breakout, Solid State Relay 등
배전반 스위치 제어 장치		Adafruit HUZAZH ESP8266 breakout, 단상 마그네트 스위치 등
냉·난방기 제어 장치		Adafruit HUZAZH ESP8266 breakout, IR LED 등
스마트빌딩 제어 웹 앱 (Web-Application)		스마트폰 (Android)
자동 출입 사용자 인증 앱		

구축된 스마트빌딩을 이용해 IoT 플랫폼의 기능을 테스트하고 검증하기 적절한 스마트빌딩 서비스를 선정 및 구체화하였다. 서비스의 자세한 내용 및 구현 방법은 Table 2와 같으며, 각 서비스는 구성 하드웨어에 공통적으로 IoT 플랫폼(IoT 서버, IoT 프록시, IoT 게이트웨이)을 포함한다.

Table 2. The list of services in the testbed

항목	서비스 내용 및 구현 방법
공통	<서비스 내용> - 모든 서비스는 외부에 위치한 사용자가 인터넷을 통하여 설정, 실행 가능
	<서비스 구성 하드웨어> - IoT 서버 : IoT 서비스 및 모니터링 제공 - IoT 프록시 : IoT 도메인 구성 및 IoT 서비스 연계 - IoT 게이트웨이 : 사물 제어용 장치 연결 - 스마트빌딩 제어 웹 앱 : 사용자가 스마트폰을 이용하여 스마트빌딩 제어
자동 출입 제어	<서비스 내용> - 출입 가능자가 출입문 주변에서 감지되는 경우 출입문을 자동으로 열어주는 서비스
	<서비스 구현 방법> - 비콘과 스마트폰을 이용하여 출입문과의 거리를 산출 - 스마트폰을 이용하여 출입문 제어 서비스 호출 - 출입 제어 서비스는 스마트폰 정보를 이용하여 출입 가능 여부 식별 및 출입문 제어
	<서비스 구성 하드웨어> - 자동 출입 사용자 인증 앱 : 비콘과의 거리를 계산하여 사용자 인증 정보를 스마트빌딩 서버에 전달 - 비콘 : 출입문 근처에 설치되어 사용자 스마트폰이 출입문과의 거리 산출을 도움 - 스마트빌딩 서버 : 출입자 인증 및 출입문 제어 서비스 호출 - 출입문 제어 장치 : IoT 게이트웨이에 출입문 제어 기능 제공
자동 절전 서비스	<서비스 내용> - 실내에 사람이 없는 경우 전원을 자동으로 제어하는 서비스
	<서비스 구현 방법> - 피플카운터를 이용한 재실 인원 감지 - 배전반 스위치를 원격으로 조작하여 절전 수행
	<서비스 구성 하드웨어> - 배전반 스위치 제어 장치 : IoT 게이트웨이에 배전반 스위치 제어 기능 제공 - 피플카운터 : IoT 게이트웨이에 재실 인원 정보 제공
실내 환경 제어 서비스	<서비스 내용> - 실내 환경을 적절하게 조절하는 서비스
	<서비스 구현 방법> - 온·습도센서를 이용한 실내 온·습도 측정 및 수집 - 온·습도에 따른 냉·난방기 자동 제어
	<서비스 구성 하드웨어> - 냉·난방기 제어 장치 : IoT 게이트웨이에 냉·난방기를 제어할 수 있는 기능(IR 리모컨 기능) 및 온·습도 정보 제공

구축된 스마트빌딩과 서비스를 이용하는 다양한 시나리오들 중 다음 시나리오를 이용한 IoT 플랫폼 기능 테스트 및 검증에 대한 구체적인 결과는 다음과 같다.

시나리오: 출입 권한이 있는 사용자가 자신의 스마트폰을 소지한 상태로 출입문 근처에 접근하면 출입문이 자동으로 열림

시나리오에 따른 IoT 플랫폼 검증 과정은 다음과 같다.

1) 출입 권한이 있는 사용자의 스마트폰에는 자동 출입 사용자 인증 앱이 설치되어 있으며, 자동 출입 사용자 인증 앱은 출입문 근처의 비콘을 인식하여 출입문과의 거리를 산출한다.

2) 출입문과의 거리가 가까워진 자동 출입 사용자 인증 앱은 스마트빌딩 서버에 출입 인증 정보를 제공하며 사용자 스마트폰과 스마트빌딩서버는 인터넷에 연결된다.

3) 스마트빌딩 서버는 IoT 서버에 등록된 출입문 제어 서비스를 호출하여 시설망에 존재하는 IoT 게이트웨이에 연결된 출입문 제어 장치를 이용하여 출입문을 제어한다.

4) 그 외에도 사용자는 Fig. 8(right)의 스마트빌딩 제어 웹 앱을 이용하여 원격에서 출입문을 제어할 수 있으며, 관리자는 Fig. 8(left)의 스마트빌딩 모니터링 웹콘솔을 이용하여 출입문 제어 및 IoT 게이트웨이 상태 정보를 모니터링할 수 있다.



Fig. 8. The monitoring web console(left) and IoT service web application(right)

시나리오를 통한 IoT 플랫폼 테스트 및 검증 과정을 통해 IoT 플랫폼에서 제공하는 시설망 지원 기능 및 IoT 서비스 제공 기능 등이 정상적으로 작동함으로써 IoT 플랫폼 테스트 및 검증 목표 수준을 충족하는 것을 확인할 수 있다.

6. 결 론

본 연구에서 개발된 IoT 플랫폼은 IoT 프록시를 이용하여 시설망을 지원함으로써 새로운 하드웨어 도입 또는 네트워크 전문가 비용을 절감할 수 있게 되었다. 또한 IoT 도메인 개념을 적용하여 IoT 기기들을 관리함으로써 좀더 효율적인 IoT 기기 관리가 가능토록 하였다.

본 연구를 통해 전문적인 지식이 없는 비전문가 또는 일반 사용자도 전문가의 도움 없이 홈 IoT와 같은 IoT 환경을 구축할 수 있으며 추가 하드웨어 도입 비용을 줄일 수 있다. 또한 소프트웨어에 대한 전문 지식이 부족한 IoT 하드웨어 개발 중소기업은 소프트웨어 전문가를 채용하지 않고 고도 IoT 환경을 구축하여 IoT 환경에 적합한 IoT 하드웨어를 제작하고 테스트할 수 있다. 따라서 본 연구는 IoT 환경 구축의 핵심 기술로써 사용될 것으로 기대되며 사설망에서 IoT 환경 구축의 어려움 해결과 IoT 환경 구축에 사용되는 비용을 절감하는데 크게 기여할 수 있다고 판단된다.

하지만 현재 IoT 플랫폼은 IoT 서버에서 사물의 정보(현재 온도, 실행 상태 등)를 저장하지 않고 있어, 필요시 항상 사물에 접속하여 해당 정보를 조회해야 한다. 이러한 한계는 IoT 서버에서 사물의 상태정보를 저장함으로써 해결이 가능할 것이다. 현재 개발한 IoT 플랫폼은 인증을 고려하지 않아 보안에 취약하다. 향후 사물의 상태 정보를 IoT 서버에 저장하는 기능, IoT 서비스 사용자를 인증하는 기능을 보완하여 다양한 분야에서 효율적이고 안전한 IoT 서비스를 제공할 수 있는 IoT 플랫폼으로 활용될 수 있을 것이다.

References

[1] Cheol-Sik Pyo, Ho-Yong Gang, Nae-Su Kim, and Hyo-Chan Bang, "IoT(M2M) technology trends and development prospects," *The Journal of the Korean Institute of Communication Sciences(Information & Communications Magazine)*, Vol.30, No.8, pp.3-10, 2013.

[2] Jae-Ho Kim, Jae-Seok Yun, Seong-Chan Choe, and Min-U Ryu, "IoT Platform Development Trend and Future Direction," *The Journal of the Korean Institute of Communication Sciences(Information & Communications Magazine)*, Vol.30, No.8, pp.29-39, 2013.

[3] S. T. Kim, J. S. Jeong, J. K. Song, and H. Y. Kim, "Trends of IoT Device Platforms and Building its Ecosystems," *Electronics and Telecommunications*, Vol.29, No.4, pp.82-90, 2014.

[4] Google.com, Google IPv6 Statics [Internet], <http://www.google.com/intl/en/ipv6/statistics.html>.

[5] Seon-Yun Kim and Ki-Young Kim, "Standardization of IoT Service Platform," *Communications of the Korea Information Science Society*, Vol.32, No.6, pp.31-36, 2014.

[6] OCEAN, OpenMobius and &Cube [Internet], <http://www.iotocean.org/main/>.

[7] Allseen Alliance, AllJoyn [Internet], <https://allseenalliance.org/>.

[8] OIC, IoTivity [Internet], <https://www.iotivity.org/>.

[9] Soo-jin, Kim, Dong-hoon, Kim, Jung-il Bae, and Seung-ho Yeon, "A Study of Controlling Information Appliances and Providing Application Services under Private Network Environment," *2007 IEEK Fall Conference*, No.1, pp.9-10, 2007.

[10] Chunglae Cho, JaeMyun KIM, and KwangRoh Park, "Controlling UPnP Devices in Private Networks from Public Networks," *Proceedings of Symposium of KICS*, pp.1487-1490, 2007.

[11] Jun Young Park and Huy Kang Kim, "A Study on the Implementation of outdoor type Virtual Private Network Gateway for Smart Grid," *Journal of the Korea Institute of Information Security and Cryptology*, pp.125-136, 2011.

[12] RFC1631, "The IP Network Address Translator (NAT)" [Internet], <https://www.ietf.org/rfc/rfc1631.txt>.

[13] JEFF TYSON, "HowStuffWorks: How Network Address Translation Works" [Internet], <http://computer.howstuffworks.com/nat.htm/printable>

[14] RFC2131, [Internet], "Dynamic Host Configuration Protocol" <https://www.ietf.org/rfc/rfc2131.txt>.



이 경 훈

e-mail : khlee@metabuild.co.kr
 2012년 한국IT전문학교 컴퓨터공학과 (전문학사)
 2013년~현재 메타빌드(주) 전임연구원
 관심분야 : 사물인터넷, 소프트웨어 공학



김 두 영

e-mail : netdoo@buall.com
 2003년 동양대학교 컴퓨터공학부(학사)
 2016년 숭실대학교 IT정책경영학과(석사)
 2010년~현재 부울(주) 수석연구원
 관심분야 : 소프트웨어 공학, 웹 공학



송 세 현

e-mail : seheon.song@metabuild.co.kr
 2001년 아주대학교 정보컴퓨터공학부(학사)
 2003년 아주대학교 정보통신공학과(석사)
 2013년 아주대학교 정보통신공학과(박사)
 2014년~현재 메타빌드(주) 책임연구원
 관심분야 : 서비스컴퓨팅, 사물인터넷, 시맨틱웹



이 상 일

e-mail : happyjoy@add.re.kr
1994년 성균관대학교 정보공학과(학사)
1996년 성균관대학교 정보공학과(석사)
2010년 성균관대학교 컴퓨터공학과(박사)
1999년~현 재 국방과학연구소 책임연구원
관심분야: 상호운용성, 사물인터넷, SOA



박 재 현

e-mail : forehand@add.re.kr
1990년 경북대학교 컴퓨터공학과(학사)
1992년 경북대학교 컴퓨터공학과(석사)
1999년~현 재 국방과학연구소 책임연구원
관심분야: M&S, SOA, 사물인터넷