LoRa 네트워크와 oneM2M 플랫폼 연동 시스템 구현

전성호, *김승구
충북대학교
shjeon@cbnu.ac.kr, *kimsk@cbnu.ac.kr

Implementation of interlocking system of LoRa network and oneM2M platform

Jeon Seongho, *Kim Seungku
Chungbuk National Univ.

요 약

본 논문은 사물인터넷 서비스에서 각기 다른 플랫폼을 하나의 공동 플랫폼으로 통합한 oneM2M 기반의 Mobius를 이용하여 센서 디바이스로부터 센싱한 데이터를 관리하는 IoT 네트워크를 구현할 때, 넓은 범위의 확장이 힘들다는 단점을 지적한다. 그리고 이를 개선하기 위해 센서 디바이스와 게이트웨이 간의 통신에 Low Power Wide Area 프로토콜인 LoRa를 사용하여 넓은 범위를 커버하는 네트워크를 구현한다. 센서 디바이스와 게이트웨이는 LoRa 프로그램을 통해 데이터를 송수신하고 수신한 데이터를 TAS와 nCube를 사용하여 oneM2M 규격의 데이터로 변환한다. 최종적으로 서버인 Mobius에서 수신하여 웹 기반의 모니터링 서비스를 통해 확인한다.

Ⅰ. 서 론

사물인터넷 서비스는 스마트홈, 스마트카, 스마트팩토리 등 다양한 영역이 존재하고 영역마다 각기 다른 서비스 플랫폼이 사용되었다. 이렇게 별도로 존재하는 서비스 플랫폼을 하나의 공통 플랫폼으로 통합하고 이종의 서비스에서 자주 사용되는 기능을 API로 제공하는 것이 공통 서비스 플랫폼의 역할이다. 하지만 이런 공통 플랫폼도 비표준 기술로 시장에 파편화되어 있었다. 이를 글로벌 표준으로 제정하여 제품과 서비스 간의 호환성을 높여 사물인터넷 시장을 확대하고 활성화 하기 위해 oneM2M이 출범하게 되었다. oneM2M 기반의 플랫폼 중 하나인 Mobius는 센서 데이터를 수집하여 관리하기 편하게 구현된 오픈소스 플랫폼으로 엔드 디바이스에 사용하는 TAS, 게이트웨이에 사용하는 nCube, 서버에 사용하는 Mobius로 구성되어 있다. 특히 서버인 Mobius는 다양한 프로토콜을 지원하여 다른 서비스와 연결하기 용이하다. 하지만 TCP/IP 모듈을 이용하여 통신하기 때문에 원거리의 센서 디바이스로부터 데이터를 받기 위해서는 다수의 게이트웨이가 필요하게 된다. 그로 인해 넓은 범위의 확장이 힘들다는 단점이 있다.

본 논문에서는 이를 보안하고자 센서 디바이스와 게이트웨이 간의 통신에 LPWAN(Low power Wide Area) 통신 프로토콜인 LoRa를 사용한다. Long Range의 약자인 LoRa는 이름처럼 장거리 통신에 강점이 있는 IoT 무선 통신 기술이다[1]. LoRa는 제한된 전력으로 기존 인프라에 쉽게 접속할 수 있고 장시간 기기를 사용할 수 있어 공통 서비스 플랫폼에 사용하기 용이하다. LoRa를 이용하여 원거리의 센서 디바이스로부터 센싱한 데이터를 게이트웨이로 받아 이를 Mobius 플랫폼으로 관리하게끔 네트워크를 구현한다.

Ⅱ. 본론

oneM2M은 스마트홈과 같은 로컬 디바이스로 구성된 서비스부터 클라우드 기반의 서비스까지 제공할 수 있다. 이를 활용하여 데이터를 클라우드에 모아 스마트카와 스마트홈 간의 메시업이 가능하고 빅데이터의 처리같은 기능을 제공할 수 있는 서비스 플랫폼이다. 2015년 1월에 요구사항, 정의, 아키텍쳐 등 10개의 표준규격을 포함하는 Release 1을 시작으로 최근에는 Release 4까지 발표되었다. oneM2M의 특징은 플랫폼이 제공하는 기능을 공통 서비스 기능으로 정의한 것이다. 공통 서비스 기능은 사물인터넷 서비스 애플리케이션에서 자주 사용되는 기능을 정의한 것으로 RESTful API를 사용한다. oneM2M의 코어 프로토콜 메시지(Primitive)는 CoAP, HTTP 및 MQTT 프로토콜 메시지를 통해 전송된다[2]. 또한, oneM2M 레퍼런스 아키텍처의 모든 엔티티는 세가지 계층으로 분화되며 각 AE(Application Entity), CSE(Common Service Entity), NSE(Network Service Entity)라고 부른다. AE는 애플리케이션 엔티티로 end-to-end 사물인터넷 솔루션을 위한 애플리케이션 로직을 제공한다. CSE는 공통 서비스 엔티티로 사물인터넷의 다양한 애플리케이션 엔티티들이 공통적으로 사용할 수 있는 기능들로 이루어진 플랫폼이다. NSE는 네트워크 서비스 엔티티로 공통 서비스 엔티티에 네트워크 서비스를 제공하며 3GGp 네트워크 연동 중심이다.

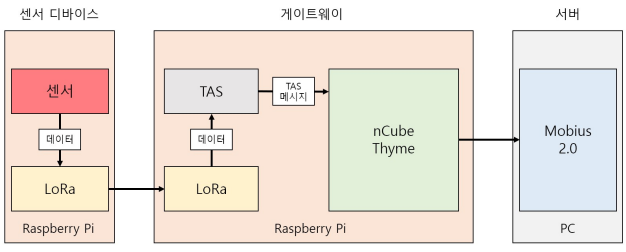

그림 1 네트워크 시스템 구성 및 동작 흐름도

본 논문의 네트워크는 oneM2M 국제 표준을 기반으로 KETI에서 개발한 Mobius 2.0 플랫폼을 이용하였다. Mobius 2.0은 IoT 서비스 제공을 위해 다양한 IoT 디바이스 정보를 관리하고, 이들 IoT 디바이스의 접근제어, 인증, 사용자 관리, 복수의 IoT 서비스 조합을 제공하는 CSE이다[3]. IoT 디바이스와 연동하기 위해 프로토콜 메시지로 HTTP, MQTT, CoAP, WebSocket을 사용한다. Mobius 플랫폼은 리소스를 생성하면 해당 리소스에 접근할 수 있는 API가 자동으로 생성되며 이를 위한 인터페이스로 REST API인 Open API를 사용한다. 그림 1과 같이 Mobius 서버에 데이터를 업로드하는 게이트웨이는 oneM2M 표준 기반 오픈소스 IoT 애플리

케이션 엔티티인 nCube Thyme이다. nCube는 TAS(Thing Adaptation Software)를 기반으로 센싱한 데이터를 수집한 다음 Mobius 서버로 측정된 값을 전달, 사물을 제어하는 기능을 한다. TAS는 데이터를 획득할 센서를 디바이스에 연결하기 위한 소프트웨어로서 센서와 nCube간의 연결통로를 만드는 역할을 한다. 먼저 사용하고자 하는 센서의 API를 TAS에서 결정한다. 그후 센서로부터 전달받은 데이터를 nCube 게이트웨이로 송신하기 위해서 TAS에서 병렬적으로 처리하고 TAS 메시지를 구성한다.

그림 2 네트워크 실험 환경

Ⅲ. 네트워크 구현

본 논문의 네트워크 구성은 다음과 같다. 센서 디바이스와 게이트웨이는 Raspberry Pi에 Libelium사의 쉴드를 사용하여 Semtech사의 LoRa 네트워크 칩인 SX1272와 연결하였다. 한국 ISM 대역인 915MHz 주파수 채널에서 실험하였고 SF는 11, Bandwidth는 125kHz로 고정한다. 센서 디바이스에는 초음파센서인 HC-SR04를 연결하여 거리 데이터를 측정한다. 게이트웨이는 nCube Thyme 2.3.2 버전을 사용하고 서버는 Mobius 2.4.36 버전을 사용한다. 네트워크의 동작 흐름은 그림 1과 같다.

센서 디바이스에서 측정한 거리 데이터는 게이트웨이에 LoRa 프로토콜을 사용하여 전송한다. 전송된 데이터는 LoRa 프로그램에서 소켓통신을 통해 게이트웨이에 구현된 TAS에 전달한다. TAS는 데이터를 병렬적으로 처리한 후 TAS 메시지를 구성하여 nCube Thyme에 전송한다. 구성된 메시지는 그림 3과 같이 JSON 형식의 데이터이다.

```
rcv:10
{"ctname":"lora","con":{"value":"TAS0, 10"}} ---->
ACK : {"ctname":"lora","con":"2001"} <----
ACK : {"ctname":"lora","con":{"value":"TAS0, 10"}} <----
rcv:10
{"ctname":"lora","con":{"value":"TAS1, 10"}} ---->
ACK : {"ctname":"lora","con":"2001"} <----
ACK : {"ctname":"lora","con":{"value":"TAS1, 10"}} <----
rcv:10
{"ctname":"lora","con":{"value":"TAS2, 10"}} ---->
ACK : {"ctname":"lora","con":"2001"} <----
ACK : {"ctname":"lora","con":{"value":"TAS2, 10"}} <----
rcv:10
{"ctname":"lora","con":{"value":"TAS3, 10"}} ---->
ACK : {"ctname":"lora","con":"2001"} <----
ACK : {"ctname":"lora","con":{"value":"TAS3, 10"}} <----
rcv:10
{"ctname":"lora","con":{"value":"TAS4, 10"}} ---->
ACK : {"ctname":"lora","con":"2001"} <----
ACK : {"ctname":"lora","con":{"value":"TAS4, 10"}} <----
rcv:10
{"ctname":"lora","con":{"value":"TAS5, 10"}} ---->
ACK : {"ctname":"lora","con":"2001"} <----
ACK : {"ctname":"lora","con":{"value":"TAS5, 10"}} <----
```

그림 3 데이터를 nCube Thyme으로 전송

TAS 메시지는 ctname과 con으로 구성되는데 ctname은 센서 디바이스의 이름을 의미하고 con은 실제로 측정한 데이터값이다. 데이터 전송이 성공했다면 nCube Thyme이 TAS에게 ACK를 보내 정상적으로 수신되었다고 알린다. 다음으로 nCube Thyme은 TCP/IP 모듈을 이용하여 MQTT 프로토콜로 Mobius 서버로 메시지를 그림 4와 같이 송신한다. 이때, nCube는 랜덤하게 생성된 rqi(request id) 값을 포함한 메시지를 보내는데 이는 MQTT 프로토콜의 요청과 응답이 비동기로 동작하기 때문에 중복메시지인지 체크하는 역할을 한다.

그림 4 nCube Thyme에서 Mobius로 데이터 전송

마지막으로 Mobius는 수신한 메시지를 웹서버로 표현함으로써 웹 기반의 실시간 리소스 모니터링 서비스를 제공한다.

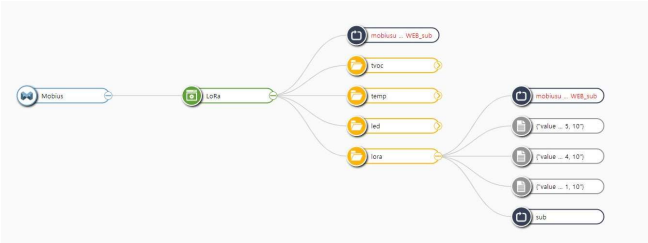

그림 5 Mobius 리소스 모니터링 웹서버

Ⅳ. 결론

본 논문은 기존의 oneM2M 기반의 Mobius 플랫폼에서 넓은 범위의 확장이 어려움을 지적하고 원거리의 센서 디바이스까지 커버할 수 있도록 LoRa 프로토콜을 이용하여 IoT 서버 네트워크를 구현하였다. 센서 디바이스와 게이트웨이는 LoRa 프로그램을 이용하여 구성했다. 게이트웨이 내에서는 수신한 데이터를 oneM2M 형식으로 송수신 가능하도록 TAS 메시지로 변환하여 nCube Thyme에 전송하고 nCube Thyme은 TCP/IP 모듈을 이용하여 MQTT 프로토콜로 Mobius 서버로 메시지를 전송했다. 최종적으로 Mobius 서버에서 모니터링 서비스를 통해 센싱한 데이터를 확인 할 수 있었다.

ACKNOWLEDGMENT

이 (성과물)은 산업통상자원부 '산업전문인력역량강화사업'의 재원으로 한국산업기술진흥원(KIAT)의 지원을 받아 수행된 연구임. (2019년 임베디드SW 전문인력 양성사업, 과제번호 : N0001884)

참 고 문 헌

[1] Heonguk Lee, "A Design of LoRa Multi-hop Network using Adaptive Spreading Factor Control Algorithm," Master Thesis, Chungbuk National University, Feb, 2019.

[2] oneM2M, "oneM2M Release 2 Technical Specifications", 2020

[3] Korea Electronics Technology Institute, "Mobius IoT Server Platform", http://developers.iotocean.org/archives/module/mobius