

MEC 환경에서 oneM2M 기반 IoT 플랫폼 등록 및 연동 구조 설계 연구

남경환, 이지은, 이지호, 정재형, 송재승*
세종대학교 정보보호학과,

nkh0721@sju.ac.kr, je.lee@sejong.ac.kr, twozio@sju.ac.kr, sj20jjh@sju.ac.kr,
jssong@sejong.ac.kr

A Study on the Design of Registration and Integration Architecture for oneM2M-based IoT Platform in MEC Environment

Kyeonghwan Nam, Jieun Lee, Jiho Lee, Jaehyung Jeong, JaeSeung Song*
Sejong Univ.

요 약

IoT 시스템에서 실시간성과 데이터 부하 분산을 위한 요구가 증가함에 따라, MEC 환경에서의 처리 구조가 중요해지고 있다. 그러나 MEC 와 IoT 플랫폼 간의 구조적 연계에 대한 실제 문서나 기술 구현은 매우 부족한 실정이다. 본 논문은 MEC 환경에서 oneM2M 기반 IoT 플랫폼을 등록 후 연동 구조를 설계하며, MEC033 API(스펙 번호 빼고, MEC 기반 표준) 기반의 등록 절차와 그 메시지 포맷을 정의하고 이를 아키텍처 수준에서 정리하였다. 제안된 구조는 향후 MEC-IoT 통합 시스템의 기반 아키텍처로 활용 가능하다.

I. 서 론

최근 IoT 시스템의 실시간성과 효율성 확보를 위한 방안으로 네트워크 엣지에서 컴퓨팅 자원을 제공하여 실시간성과 지연 최소화를 지원하는 구조인 Multi-access Edge Computing (MEC) 기반의 구조가 주목받고 있다. 기존의 중앙 집중형 클라우드 구조는 지연(latency) 및 데이터 부하 문제로 인해 다양한 산업 분야에서 적용에 한계를 보이고 있다.

한편, oneM2M 은 IoT 디바이스 간의 상호운용성과 리소스 관리를 표준화하기 위해 개발된 글로벌 IoT 플랫폼 아키텍처이다. 그러나 oneM2M 은 일반적으로 클라우드 기반으로 동작하며, 실시간 데이터 처리 및 분산 제어 측면에서 한계를 가진다. 이를 보완하기 위해 oneM2M 플랫폼을 MEC 환경과 연동하여, 엣지 단에서 플랫폼 등록과 데이터 처리를 수행할 수 있는 구조가 요구된다.

그러나 이러한 구조적 통합에 대한 기술적 문서나 아키텍처 수준의 설계 시나리오는 매우 제한적이다. 기존 연구들은 MEC 기반 시스템에서의 자원 할당과 오프로딩, 네트워크 최적화 등에 집중되어 있으며, 실제 IoT 플랫폼의 등록 절차나 통신 흐름과 같은 구조적 연계에 대한 논의는 부족한 실정이다 [1][2].

본 논문에서는 이러한 공백을 해소하기 위해, IoT 플랫폼 등록 및 테스트를 위한 가상의 MEC 실행 환경에 해당하는 MEC Sandbox 와 oneM2M 기반 IoT 플랫폼 사이의 연동 구조와 등록 절차를 설계하였다.



그림 1 - MEC 에 등록된 IoT 플랫폼, 디바이스 연동 구조

구체적으로는 MEC 환경에서 IoT 플랫폼의 등록과 데이터 전송 및 관리를 명세한 ETSI MEC033 IoT API 를 기반으로, IoT 플랫폼 등록을 위한 메시지 포맷과 통신 흐름을 정의하고, 이를 아키텍처 수준에서 구성하였다 [3]. 제안된 구조는 MEC 와 IoT 플랫폼 간 실시간 연동의 가능성을 제시하며, 향후 엣지 기반의 IoT 시스템 개발의 기반이 될 수 있다.

본 논문의 구성은 다음과 같다. 2 장에서는 본 시스템의 전체 구조와 연동을 위한 IoT 플랫폼의 MEC Sandbox 로의 등록 절차를 설명한다. 마지막으로 3 장에서 결론 및 향후 연구에 대해 논한다.

II. 본론

본 단락에서는 MEC 환경에서 IoT 플랫폼 등록 절차와 전체 구조를 다루며 이는 MEC033 API 문서에 기반하여 설계되었다.

2.1 전체 구조

이 단락에서는 oneM2M 플랫폼이 MEC Sandbox 에 IoT 플랫폼으로 등록되어 연동이 될 때의 전체 시스템 구조를 엣지단을 중심으로 설명한다. (그림 1)은 센서가 생성한 데이터를 엣지단의 MEC 플랫폼이 수신하고, MEC 플랫폼에 등록된 oneM2M IoT 플랫폼이 MEC App 또는 기타 응용 서비스와 연동될 수 있는 구조를

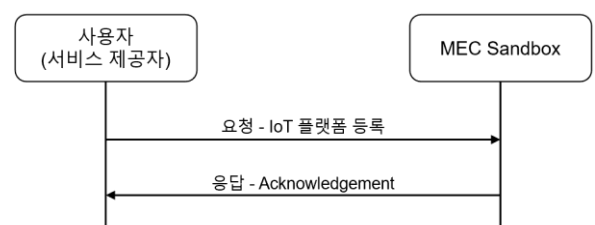


그림 2 - IoT 플랫폼 등록 절차

```

{
  "IoTPlatformInfo": {
    "iotPlatformId": "oneM2M tinyiot",
    "userTransportInfo": [{
      "id": "tinyiot HTTP",
      "name": "tinyiot HTTP protocol",
      "description": "tinyiot HTTP description",
      "type": "REST_HTTP",
      "protocol": "HTTP",
      "version": "1.1",
      "endpoint": {
        "addresses": [{"host": "192.168.1.100",
          "port": 8080}],
        "security": {"":""},
        "implSpecificInfo": {"":""}
      }
    ]
  },
  "enabled": true
}

```

리스트 1 - IoT 플랫폼 등록 메시지 예시

시각화한 것이다. (그림 1)과 같이 엣지에 IoT 플랫폼이 등록되어 연동 구조가 형성되면, 센서에서 생성된 데이터를 엣지에 등록된 oneM2M 기반 IoT 플랫폼에 전송할 수 있게 되며, 이때 IoT 플랫폼은 데이터를 수신하여 내부적으로 리소스 트리에 반영, 이를 통해 해당 디바이스의 상태, 측정값, 이벤트 정보 등을 플랫폼 상에 지속적으로 갱신한다.

MEC 에 등록된 IoT 플랫폼은 MEC Sandbox 상에서 공식적인 IoT 플랫폼으로 인식되어 MEC 내부 또는 외부의 응용 서비스가 oneM2M 플랫폼으로부터 리소스 정보 또는 데이터를 참조할 수 있다. 이러한 구조는 센서에서 엣지에 등록된 oneM2M 플랫폼으로, 최종적으로 클라우드를 통해 서비스 계층으로 이어지는 전형적인 엣지 아키텍처의 흐름을 기반으로 한다.

2.2 등록 절차.

이 단락에서는 oneM2M 기반 IoT 플랫폼의 MEC Sandbox 로의 등록 절차를 설명한다. 1 절에서 설명한 MEC Sandbox 와 oneM2M 플랫폼 사이의 연동 구조를 위해서는 먼저 qusru 해당 플랫폼을 IoT Platform 으로 등록하는 과정이 선행되어야 한다. (그림 2)는 사용자가

서비스 제공자로서 MEC Sandbox 에 oneM2M 플랫폼을 IoT 플랫폼으로서 등록 요청을 보내고 응답을 받는 절차를 시퀀스 다이어그램 형태로 나타낸 것이다. 등록 요청은 사용자가 구성한 IoT 플랫폼 정보를 포함한 메시지를 기반으로 이루어지며, 등록에 성공하면 MEC Sandbox 는 이에 대한 응답으로 acknowledgment 를 반환한다. 이때, 등록 요청 메시지는 ETSI MEC033 표준에서 정의된 데이터 모델을 따라야 하며, 이에는 사용 프로토콜 정보나 보안 설정 등 IoT 플랫폼의 정보가 명시되어야 한다.

(표 1)은 MEC033 표준에서 요구하는 IoT 플랫폼 등록 시 명세해야 하는 필수 데이터 필드를 정리한 것이다. 각 필드는 플랫폼 식별자, 통신 프로토콜 식별자와 그 정보, 보안 정책, 엔드 포인트, 토픽 정보 등으로 구성되어 있으며, 이러한 정보를 기반으로 한 전체 등록 메시지 포맷 예시는 (리스트 1)에 제시하였다. 이 메시지는 실제 사용자(서비스 제공자)에 의해 구성되며, MEC Sandbox 에 HTTP POST 요청으로 전달된다. oneM2M 플랫폼이 192.168.1.100 의 IP 주소와 8080 port 번호를 포함한 엔드 포인트 정보와 사용하는 프로토콜 정보, 보안 정보 등을 포함한 필수적인 플랫폼 정보를 담은 IoT platform registration 요청의 전체 메시지 포맷 예시이다.

III. 결론

본 논문에서는 MEC Sandbox 환경에서 oneM2M 기반 IoT 플랫폼의 등록과 전체 구조를 설계하였다. 등록 절차는 ETSI MEC033 IoT API 기반의 데이터 구조와 메시지 포맷에 따라 구성되며, 실제 MEC Sandbox 환경을 기반으로 설계 시나리오를 구성하였다. 본 구조는 MEC 와 IoT 플랫폼 간 실시간 연계를 구성하기 위한 초기 구조로 활용 가능하며 향후 실시간성이 요구되는 스마트 팩토리 또는 스마트 시티의 센서 모니터링 시스템과 같은 실제 시스템 설계에 확장 적용이 가능할 것이다. 또한 보안 처리 모듈과의 연계나 양방향 이벤트 기반 처리 및 엣지-클라우드 간 연동 구조의 최적화 방향으로도 연구가 필요하다.

ACKNOWLEDGMENT

이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원-대학 ICT 연구센터(ITRC)의 지원을 받아 수행된 연구임(IITP-2025-RS-2021-II211816). 교신 저자 (송재승 교수).

참 고 문 헌

- [1] W. Wu, W. Feng, Y. Fang, and Z. Lin, "Multi-HAP-Assisted Computation Offloading in Space-Air-Ground-Sea Integrated Network," IEEE Internet of Things Journal, vol. 12, no. 4, pp. 987-1001, Feb. 2025.
- [2] L. H. N. Rodrigues, C. F. M. Almeida, N. Kagan, and L. H. L. Rosa, "Co-Simulation of Interconnection Between Smart Power Grid and Smart Cities Platform via mMTC," Sensors, vol. 25, no. 5, 1517, Mar. 2025.
- [3] ETSI, "Multi-access Edge Computing (MEC); IoT API," ETSI GS MEC 033 V3.1.1, ETSI, Dec. 2022.

표 1 - MEC033 IoT 플랫폼 정보 필수 데이터 필드

Name	Type	Description
iotPlatformID	String	Iot 플랫폼 식별자
id	String	Transport 식별자
name	String	Transport 이름
type	TransportType	Transport 방식
protocol	String	통신 프로토콜
version	String	프로토콜 버전
security	SecurityInfo	보안 관련 정보
endpoint	EndpointInfo	Iot 플랫폼 주소
implSpecificInfo	Structure	토픽 명세 정보