IoT 가상환경 플랫폼에서의 무결성 보장 시스템:Hyperledger Indy와 MQTT를 통하여

(Integrity Guarantee System in IoT Virtual Environment Platform: Through Hyperedfger Indy and MQTT)

홍유성*, 김근형**

(Yoosung Hong, Geun-Hyung Kim)

요 약

본 논문에서는 Hyperledger Indy와 MQTT를 결합하여 가상환경에서의 IoT(Internet of Things) 디바이스의 데이터 무결성을 높이는 시스템을 제안한다. 이 시스템은 발행-구독(pub/sub) 패턴의 통신에서 분산형 네트워크를 활용한 DPKI(Decentralized Public Key Infrastructure) 구조를 실현하여 중앙집중형 시스템의 한계를 보완한다. IoT 디바이스의 데이터 무결성을 보장하기 위해 디지털 서명 기술을 적용하였고 클라이언트, IoT 디바이스, 브로커, 블록체인의 네 가지 핵심 요소 간의 통신 시나리오와 분산 식별자(Decentralized Identifier)를 활용한 토픽 구조를 통해 가상 환경에서 안전화고 투명한 데이터 교환을 위한 체계적인 방법을 제시한다. 본 논문은 제안된 시스템의 성능을 입증하기 위해 네 가지 시나리오에 대해서 실험을 수행하고 가상환경에서의 통신 성능을 평가하였다. 실험 결과 제안된 시스템이 가상환경에서 신뢰성 있는 IoT 데이터 통신 구조를 제공함을 확인하였다.

■ 중심어 : 가상환경 ; 사물인터넷 ; 블록체인 ; MQTT ; 분산 식별자

Abstract

In this paper, we propose a system that improves the data integrity of IoT(Internet of Things) devices in the virtual environment by combining Hyperledger Indy and MQTT(Message Queuing Telemetry Transport). The system complements the limitations of the centralized system by realizing a DPKI(Decentralized Public Key Infrastructure) structure that utilizes a distributed network in publish-subscribe(pub/sub) pattern communication. Digital signature technology was applied to ensure the data integrity of IoT devices and communication scenarios between the four core components of the client, IoT device, broker, and blockchain, as well as a topic structure using a decentralized identifier to ensure safety in the virtual environment. We present a systematic method for transparent data exchange. To prove the performance of the proposed system, this paper conducted experiments on four scenarios and evaluated communication performance in a virtual environment. The experimental results confirmed that the proposed system provides a reliable IoT data communication structure in a virtual environment.

■ keywords : Virtual Environment ; Internet of Things ; Blockchain ; MQTT ; Decentralized Identifier

Ⅰ. 서 론

메타버스는 현실 세계의 다양한 물질적, 사회적 요소를 모방한 가상환경에서 현실의 제약을 벗어나 소통, 협업, 창작 등 다양한 활동을 할 수 있도록 제공된 플랫폼이다[1]. 이러한 메타버스는 현실 세계와 디지털 세계를 연결하는 대표적인 기술인 IoT(Internet of Things)와 밀접한 관련

* 학생회원, 동의대학교 게임공학과

** 정회원, 동의대학교 게임공학과, 불록체인기술연구소

이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2021R1F1A1047573).

접수일자 : 2024년 01월 08일 게재확정일 : 2024년 02월 17일

수정일자 : 2024년 02월 09일 교신저자 : 김근형 e-mail : geunkim@deu.ac.kr

이 있다. 메타버스에서의 IoT는 부착된 센서를 통해 현실 세계에서 측정한 데이터를 가상환경 속의 사용자에게 전송하여 사용자가 현실 세계의 정보를 알 수 있게 해주는 역할을 한다[2]. 이렇게 전송된 데이터는 사용자에게 직관적이고 풍부한 정보를 제공하여 가상환경 속에서의 상호작용을 더욱 현실적으로 만든다[3].

메타버스와 IoT가 결합한 환경에서 메시지 무결성은 핵심적인 중요성을 갖는다. 이 환경에서는 다양한 디바이스와 시스템이 연결되어 실세계 데이터를 가상 세계로 전송하고 공유하여 현실 세계에서의 의사소통과 제어에 영향을 미치게 된다. 메시지의 무결성은 데이터의 정확성과 신뢰성을 보장하는 중요한 요소로 이를 통해 가상환경에서 수집된 IoT 데이터가 변조되지 않았음을 확신할 수 있다. 무결성이 보장되지 않으면 메타버스에서의 신뢰성 있는 의사소통 및 제어를 위협할 수 있기 때문에 이를 해결하기 위한 효과적인 메커니즘과 보안 기술의 적용이 필수적이다.

본 논문에서는 메타버스와 IoT가 결합한 환경에서 메시지 무결성을 확보하기 위해 MQTT (Message Queuing Telemetry Transport) 프로토콜과 탈중앙화된 식별자인 DID(Decentralized Identifier)를 활용하여 기존의 중앙집중식 PKI (Public Key Infrastructure)를 탈중앙화한 DPKI (Decentralized Public Key Infrastructure)에 기반한 통신 시스템을 제안한다. 이를 통해 데이터 무결성을 보다 효과적으로 검증할 수 있다. 이 시스템은 MQTT 프로토콜의 발행-구독(pub/sub) 패턴을 기반한 데이터 전송 메커니즘에 DID 기술을 결합하여 사용자와 디바이스가 중앙인증기관에 의존하지 않고 고유하게 식별하여 주고받는 데이터의 무결성을 보장한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 연구와 관련한 기술과 기존의 연구를 분석한다. 3장에서는 2장에서 분석한 기술과 관련 연구를 참고하여 메시지의 무결성을 보장하는 시스템을 제안 및 설계한다. 4장에서는 제안된 시스템을 구현하고 성능을 평가한다. 5장은 결론으로, 앞으로의 연구 방향을 서술한다.

Ⅱ. 관련 기술 및 연구 분석

1. MQTT 프로토콜

MQTT 프로토콜은 발행-구독(pub/sub) 패턴을 기반한 경량 메시징 프로토콜로 헤더와 페이로드로 구성된 MQTT 메시지(그림 3)형식으로 IoT 통신에 널리 사용되고 있다. MQTT 통신 구조는 그림 1과 같이 브로커(Broker)와 디바이스, PC, 모바일 디바이스 등으로 구성된다. 브로커는 중앙 서버 역할을 하고 디바이스, PC, 모바일 디바이스는 클라이언트 역할을 한다[4].

MQTT 프로토콜의 메시지는 제어정보를 포함하는 헤더(Header)와 실제 전송할 데이터가 포함되는 페이로드(Payload)로 구성된다(그림 3). MQTT 프로토콜의 토픽은 메시지의 종류(주제)에 해당하는 것으로 메시지를 발행하거나 구독할 때 설정되는 것이다. 토픽은 메시지가 속하는 주제를 정의하는 것으로 계층적인 구조를 가지고 각 레벨은 슬래시(/)로 구분한다. 토픽의 표현 예인 /home/living-room/temperature은 home 내 living-room의 temperature를 뜻한다.

메시지 발행(publish)은 클라이언트가 특정 토픽의 데이터를 브로커에 전송하는 과정으로, 발행자(Publisher)가 브로커에게 메시지를 보내고 브로커는 해당 토픽을 구독 중인 구독자(Subscriber)에게 메시지가 전달된다. 발행자가 설정한 토픽과 클라이언트의 구독하는 토픽 정보는 브로커에서 관리하고 메시지 라우팅을 위한 효율적인 기능을 제공한다. MQTT 구조에서 사용자는 특정 토픽의 존재 여부를 사전에 알 수 없으며, 사전에 정의된 토픽 명세나 개발자가 제공하는 문서를 통해 토픽의 구조와 목록을 확인하여야 한다[4]. 본 논문에서는 토픽 정보를 블록체인에 저장하고 블록체인을 통해 토픽 정보를 얻도록 제안하

였다.

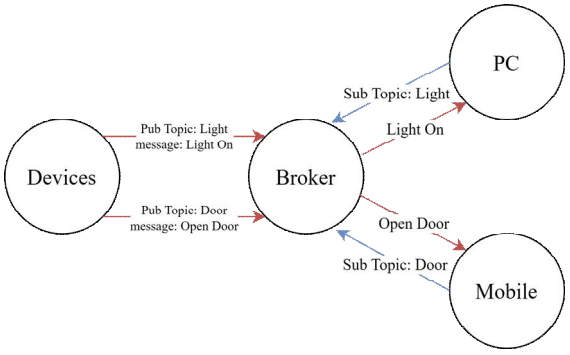

그림 1. MQTT 통신 구조

2. PKI

PKI는 공개 키와 개인 키를 사용하여 보안 기능을 제공하는 구조이다. 사용자는 수학적인 연관성이 있는 공개 키와 개인 키 쌍을 가진다. 개인 키로 암호화된 데이터를 공개 키로 복호화하거나, 공개 키로 암호화된 데이터를 개인 키로 복호화할 수 있다. 공개 키는 모두에게 공개된 키이며 송신자가 수신자에게 암호화 메시지를 보낼 때 수신자의 공개 키를 사용할 수 있다. 암호화된 메시지는 수신자만이 알고 있는 개인 키로만 복호화할 수 있어 다른 사용자는 메시지의 내용을 해독할 수 없다. 키 쌍은 '디지털 서명'에도 사용된다. '디지털 서명'은 메시지의 무결성을 위해 사용되는 암호화 기술이며 송신자는 자신의 개인 키로 메시지에 서명을 생성하고, 수신자는 서명 데이터와 원본 데이터, 송신자의 공개 키로 메시지의 무결성을 검증한다[5].

현재의 MQTT 프로토콜은 암호화나 인증 메커니즘이 내장되어 있지 않아, 허가되지 않은 사용자의 디바이스 접근에 대한 보안이 취약하다. MQTT 프로토콜을 제안한 OASIS는 이러한 취약성을 보완하기 위해 TLS/SSL을 도입하는 것을 권장한다. TLS/SSL은 사용자의 신원 인증과 인증서 관리를 중앙인증기관인 CA(Certificate Authority)에 의존하는 PKI 구조를 사용하고 있다. 이러한 PKI는 중앙화된 키 및 인증서 관리로 사용자는 서비스 이용에 집중할 수 있지만, 중앙인증기관의 신뢰성 문제, 해킹 위험, 단일 실패 지점(SPOF:Single Point of Failure)과 같은 문제가 발생할 수 있다[6].

3. DPKI

DPKI는 기존 PKI의 한계를 극복하고자 하는 목적으로 개발되었다. DPKI는 분산형 네트워크에서 활용되는 공개 키 인프라를 제공한다. 중앙인증기관을 사용하는 PKI의 문제를 완화하면서 사용자는 자체적으로 공개 키를 생성하고 관리할 수 있는 구조를 가진다.

DID는 블록체인과 같은 분산형 네트워크에서 식별자를 제공하는 역할을 한다. W3C에서 표준이 이루어진 DID는 중앙인증기관과 무관하게 사용자가 생성하는 특별한 종류의 식별자로, 블록체인을 기반으로 한 탈중앙화, 영속성, 암호 기술로 검증이 가능하다는 특징을 가지고 있다. 이 기술은 중앙인증기관 없이 개인정보 보호, 신원 인증, 거래 처리 등을 수행할 수 있다[7][8].

DID 기술은 DID 문서(DID Document)에 기록된 정보로 구성된다. DID 문서는 사용자의 공개 키, 서비스 종단점, 인증 방법 및 기타 중요한 메타데이터를 포함하며 이 정보들은 분산된 형태로 저장되어 중앙기관의 개입 없이 DPKI 구조 내에서 DID를 통한 신원 인증 및 무결성 검증을 수행할 수 있다. DID 문서의 주요 구성요소는 표 1과 같다[9].

표 1. DID 문서의 구성요소

속성	설명
ID	해당 구성요소를 통해 식별될 객체의 DID가 포함
Public Key	DID 소유자의 공개 키, 디지털 서명 및 암호화를 위한 키 교환에 사용
Service Endpoint	DID 소유자가 제공하는 서비스의 위치를 나타내는 URL 또는 엔드포인트
Authentication Method	DID의 소유자가 해당 식별자와 관련된 인증 방법을 제공하여 자신의 신원을 증명할 수 있도록 한다

또한 PKI와 DPKI의 주요 특징의 비교는 표 2와 같다[9].

표 2. PKI와 DPKI의 주요 특성 비교

특성	PKI	DPKI
모델	중앙집중식	분산/탈중앙화
신원 증명 방식	중앙인증기관이 사용자의 신원을 검증	분산 네트워크에 저장한 개인 식별 정보를 통해 사용자가 직접 검증
키 배포	중앙기관	개인
안정성	중앙기간의 취약성 고려	분산된 구조로 인한 향상된 보안성

이러한 DID 기술을 활용한 DPKI의 특징은 메타버스에서 무결성 보장 시스템을 구현하는 데 핵심적인 역할을 한다. 블록체인의 투명성과 불변성은 시스템에 저장된 데이터에 대한 높은 신뢰성과 무결성을 제공하며 DID를 통해 각 사용자는 고유한 주체로서 메타버스 내에서 신원을 증명할 수 있다. 이는 메타버스에서 IoT 데이터가 변조될 수 있는 위험으로부터 사용자를 안전하게 보호하는 데 기여한다[10].

4. 관련 연구

MQTT 프로토콜은 경량의 프로토콜로 자원이 제한된 환경을 위해 설계되어 중요하지 않은 데이터를 전송하는데 충분한 기본 보안 조치만 고려하였다. 그러나 중요한 데이터를 전송하는 경우에는 별도의 보안 메커니즘이 요구된다. 이에 따라 MQTT 통신 구조의 디바이스, 사용자의 보안 인증 메커니즘에 대한 연구가 진행되었다. [11]에서는 장치의 인증을 위해 이더리움 블록체인 기반 2단계 대역외 채널을 구현하는 MQTT용 새로운 OTP(One Time Password) 인증 방법을 제안하였다.

Abubakaret[12]는 사용자, 모바일 단말 및 IoT 디바이스의 DID를 블록체인에 공개적으로 등록해야 하는 MQTT 디바이스에 대한 인증 및 권한 부여 방법을 제안하였다. 이 논문의 저자는 통신 채널 보호 방법에 대해 자세히 설명하지 않았다.

Dixit[13]은 분산형 IoT 프레임워크를 정의하고, 검증가능 자격증명(VC)과 디바이스 URL을 블록체인에 공개하고 자격증명 교환으로 디바이스 간 인증과 권한 부여를 한다. 이는 자격증명이 정보보호가 필요한 요소임에도 블록체인을 통해 모두에게 공개하는 문제가 있고 메시지 기밀성과 신뢰성을 달성하는 방법에 대해서 자세히 설명하지 않았다.

Fotiouet[14]는 사용자가 권한 부여 서버에 인증하고 개인정보를 보호하는 DID와 VC를 수신하여 IoT 디바이스에 대한 역량 기반 접근 제어 시스템을 제안했다. VC에는 IoT 장치에 대한 권한이 부여된 사용자의 작업을 정의한다. 디바이스는 소유자에 의해 관리되며 신뢰할 수 있는 발급자 목록으로 구성되어 새로운 발급자가 시스템에 가입하면 모든 장치를 갱신해야 하는 문제점이 있다.

MQTT 프로토콜과 DID 기술을 결합한 인증 메커니즘이 [15]에서 제안되였다. 본 논문의 저자는 발행자, 구독자, 브로커 외에 토픽 관리자를 정의하며 토픽 관리자가 토큰 생성과 인증 작업을 수행하여 블록체인에 저장하고 토픽에 대한 사용자를 인증하고 있다. 브로커는 토큰을 검증하기 위해 키 교환 알고리즘을 수행한다. 이를 통해 DID와 토큰을 기반으로 개체를 식별하고 권한을 부여하여 검증된 사용자만이 MQTT 프로토콜을 통해 데이터를 주고받을 수 있도록 하였다. 이 논문은 토큰 기반의 토픽에 대한 접근 제어를 하는 것으로 메시지의 신뢰성 보장 방안을 다루지 않았다.

Philipp[16]는 DID, VC를 기반으로 개인정보 보호를 위한 질의-응답(Challenge-Response) 형식의 인증 및 권한 부여 체계를 제안하여 동적으로 변경되는 시스템의 참가를 지원하는 것으로 DID 메소드로 did:web을 고려하고 있어 DID 문서의 불변성을 보장할 수 없는 문제가 있다.

기존의 연구들은 IoT 생태계의 IoT 디바이스와 사용자 인증 및 권한 메커니즘을 중점적으로 연구한 것으로 통신채널 보호와 메시지의 신뢰성을 달성하는 방법에 대해서 상세하게 논의되어

있지 않다.

본 논문에서는 DID와 VC가 기반한 자기주권 신원(Self-Sovereign Identity) 생태계에서 MQTT 프로토콜을 활용하여 IoT 디바이스와 클라이언트 간 메시지 신뢰성을 확보하는 방법을 연구한다. MQTT 프로토콜을 구성하는 IoT 디바이스, 클라이언트, 브로커 모두 DID를 생성하고 Hyperledger Indy로 구성된 블록체인에 등록하고 블록체인에 저장된 DID 문서에 기반하여 피어의 서비스 종단점 정보와 DPKI에 기반한 메시지 신뢰성 보장 메커니즘을 연구한다.

Ⅲ. 제안된 기술

본 연구에서는 DID와 MQTT 프로토콜을 활용하여 통신채널의 보안성과 IoT 데이터의 무결성을 강화하는 시스템을 제안한다. 제안된 시스템의 핵심 통신 시퀀스는 그림 2와 같다. 이 구성을 통해 DID에 기반한 자기주권 신원 생태계에서 데이터 전송의 신뢰성을 보장하는 통신 환경을 제공하는 것이 본 연구의 목표이다.

해당 시퀀스를 이루는 주체는 클라이언트, 디바이스, 블록체인, 그리고 브로커가 있다. 클라이언트는 사용자가 접속한 가상환경을 의미하며 디바이스는 사용자에게 센서 데이터를 전송하는 역할을 한다.

클라이언트, IoT 디바이스, 브로커는 실행 시 독자적인 로컬 지갑을 생성하여 블록체인 네트워크에 사용한 DID와 공개 키, 암호화 알고리즘 정보, 서비스 종단점 등의 정보를 저장한다. 각 IoT 디바이스는 브로커에게 연결 시 데이터를 발행할 토픽에 자신의 DID를 추가한 정보를 브로커에게 전송하여 브로커에 저장한다.

각 토픽 정보는 {디바이스의 DID}/{센서 이름}의 계층 형태로 되어 있으며 클라이언트에게 DID 정보와 센서에서 생성되는 데이터와 관련하여 발행될 토픽을 명시한다. 클라이언트는 제공된 브로커와 연결 시 브로커로부터 디바이스가 저장한 토픽 정보를 요청할 수 있으며 원하는 토픽을 구독 및 해지할 수 있다.

디바이스는 센서로부터 수집한 데이터에 개인키를 활용하여 디지털 서명을 한 후 {디바이스의 DID}/{센서 이름}의 토픽으로 원본 데이터와 서명 데이터를 페이로드에 담아 발행한다. 이를 수신한 브로커는 해당 토픽을 구독한 모든 클라이언트에게 메시지를 전송한다. 클라이언트는 브로커로부터 수신한 서명 데이터와 원본 데이터에서 얻은 디바이스의 DID를 블록체인에서 조회하여 DID 문서로부터 공개 키를 얻어 메시지의 무결성을 검증하여 신뢰성을 강화한다. 해당 구조에서 사용되는 메시지 포맷은 그림 3과 같다.

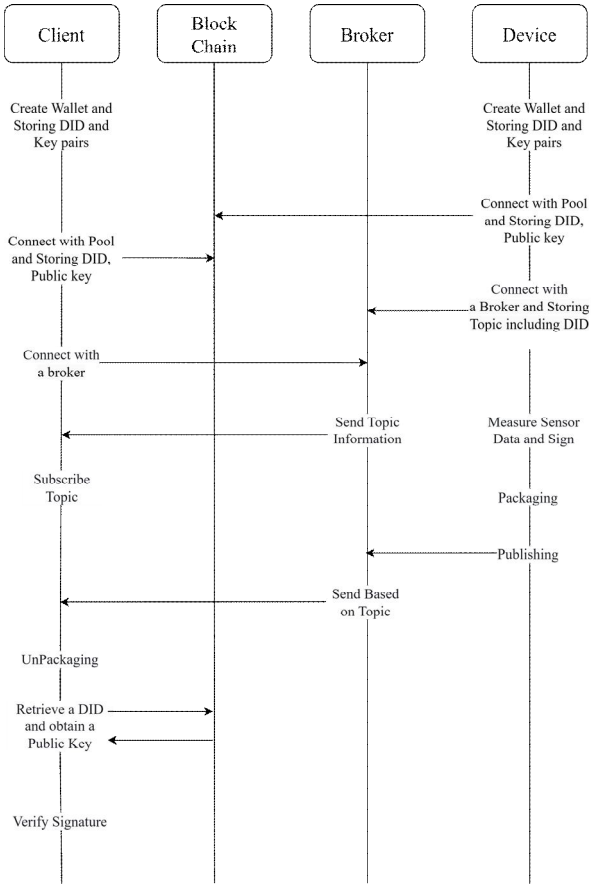

그림 2. 제안된 통신 시퀀스

		7	6	5	4	3	2	1	0
Fixed Header	byte 1	Control Packet Type				Flasgs			
	byte 2 ~ n (1 ~ 4byte)	Remaining Length							
Variable Header	byte n+1 ~ m	Topic Name							
Payload	byte m+1 ~	Sign Data + Origin Data							

그림 3. 메시지 포멧

Ⅳ. 구현 및 성능 평가

본 장에서는 제안된 시스템의 성능을 평가하기 위한 시스템 구현 내용 및 다양한 실험 시나리오에 대한 결과를 분석한다. 구현에 사용된 도구 및 라이브러리는 표 3에 나타내고 스마트홈 가상환경과 브로커를 개발하기 위해 사용한 컴퓨터 사양은 표 4에 나타냈다. 또한 본 논문에서 제안한 시스템을 구현한 소프트웨어는 github 저장소 [17]에 공개되어 있다.

표 3. 구현에 사용된 도구 및 라이브러리

구분	도구 및 라이브러리
가상환경	Unity, Paho MQTT Library[18]
블록체인	Hyperledger Indy[19]
브로커	MqttNet[20]
통신 프로토콜	MQTT
IoT 디바이스	RaspberryPi4, Paho MQTT Library[21]
서명 알고리즘	Ed25519

표 4. 컴퓨팅 사양

운영체제	Windows11
CPU	Intel(R) Core(TM) i7-9700
RAM	32GB
GPU	NVIDIA GeForce GTX 2070

1. 구현 과정

실험의 가상환경은 Unity 2022.3.29.f를 사용하여 현실 세계의 스마트 홈을 구현하고 물리 공간의 센서로부터 수집된 데이터를 스마트 홈의 디스플레이에 표시하고 사용자는 그림 4에 보인 아바타를 이동하면서 가상환경과 연결된 센서에서 측정한 센서 데이터 취득 및 제어를 하도록 구현하였다.

IoT 디바이스는 RaspberryPi4 보드를 사용하였다. RaspberryPi4 보드는 다양한 센서 연결 옵션을 제공하며 연결된 센서를 통해 현실 세계의 데이터를 수집 및 처리할 수 있다. 본 실험에서 구현한 RaspberryPi4 보드에는 물체의 거리를 측정하는 초음파 센서와 LED 전구, 온습도 센서가 연결되어 있다.

그림 4. 가구, 아바타, 센서가 배치된 스마트홈 가상환경

IoT 디바이스는 센서로부터 측정된 물체의 거리 데이터, 온습도 데이터 또는 LED 전구의 전원 데이터를 자기 개인 키로 서명하고 별도의 토픽으로 브로커에게 전송한다. 클라이언트는 토픽을 구독하여 수신할 센서 데이터를 선택할 수 있다. 그림 5는 실험에 구현한 RaspberryPi4 보드와 센서를 보인다.

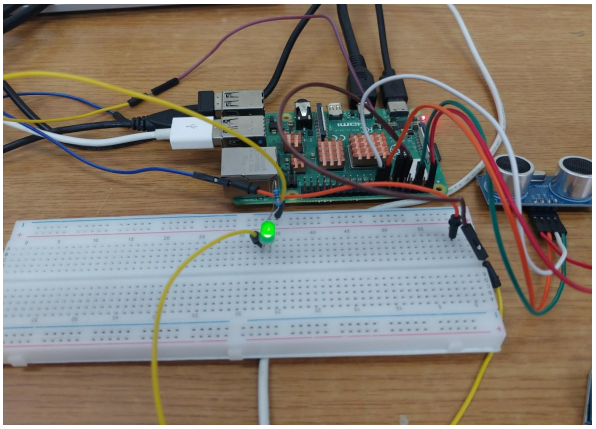

그림 5. 실험에 구현된 RaspberryPi4 기반 IoT 디바이스

실험 환경에 사용된 토픽의 정의는 표 4와 같다. 표 4의 테이블의 {device DID} 영역에는 IoT 디바이스의 DID 값을 기재한다. 디바이스는 해당 센서로부터 수집된 데이터를 디바이스의 DID와 하위레벨의 센서 이름으로 토픽을 발행한다.

표 4. 실험을 위해 정의된 토픽 이름

센서	토픽
초음파 센서	home/devices/{device DID}/ultrasonic
LED 전구	home/devices/{device DID}/led
온습도 센서	home/devices/{device DID}/tem

블록체인은 Hyperledger Indy로 구현하였고 Node Pool 형태로 존재한다. 클라이언트와 디바이스는 분산 네트워크의 정보가 포함된 제네시스 파일을 이용하여 동일한 Node Pool에 접속하여 DID와 공개 키를 저장한다.

클라이언트와 디바이스는 실행 시 Node Pool에 연결한 후 DNS(Domain Name System)에 등록된 브로커에게 연결을 시도한다. 연결이 성공적으로 수행되면 IoT 디바이스는 연결된 센서를 기반한 토픽 정보를 브로커에 저장한다. 사용자가 접속한 가상환경도 브로커에게 연결되어 구독할 수 있는 토픽 정보를 취득한다.

IoT 디바이스, 클라이언트가 MQTT 브로커와 DID 기반 블록체인인 Hypeledgr Indy와 연동하기 위해서 Paho MQTT Library와 Hyperledger Indy SDK를 사용하였다.

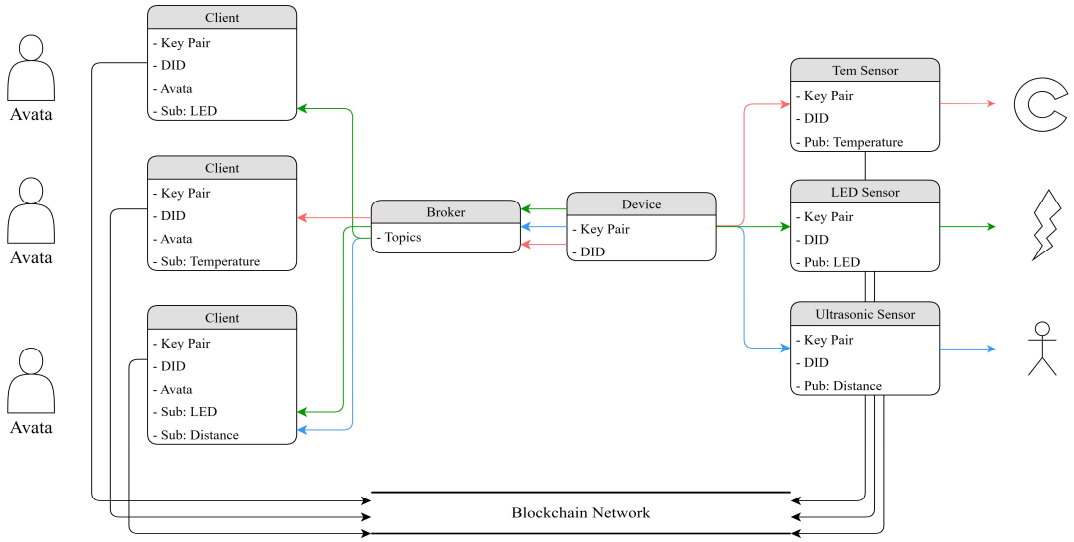

그림 6. 사용자가 가상환경을 통해 센싱 데이터를 확인하는 과정

데이터 무결성을 보장하기 위해 Hyperledger Indy SDK를 이용하여 센서 데이터와 센서 데이터의 서명을 JSON(JavaScript Object Notation) 형식으로 패키징하여 브로커로 전달한다. 브로커로부터 이를 수신한 클라이언트는 이를 통해 토픽에 명시된 DID와 함께 무결성이 검증된 센서 데이터만 가상환경에 표출하여 사용자가 확인할 수 있다.

그림 6에는 사용자가 가상환경에서 구현된 모델의 전체 모습으로 센서 데이터를 확인하는 과정을 보인다. 그림 7은 사용자가 초음파 센서와 LED 전구의 토픽을 구독하여 수신된 센서 데이터를 확인한 모습을 보인다. 그림 8은 VonNet에서 확인된 디바이스의 DID와 DID의 공개 키를 보인다.

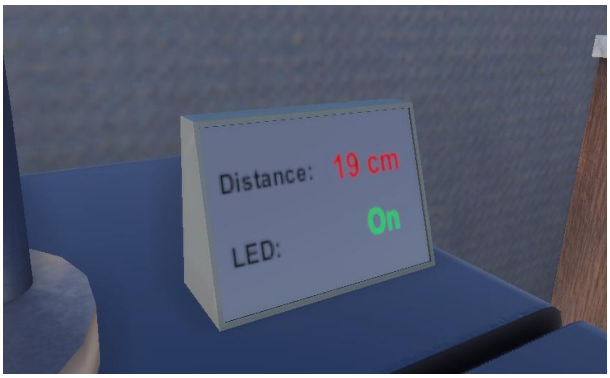

그림 7. 가상환경에서 확인한 IoT 디바이스의 센서 데이터

그림 8. VonNet에서 확인된 디바이스의 DID와 공개 키

2. 성능 평가

본 연구에서는 IoT 디바이스에서 생성된 센서 데이터의 무결성 검증과 통신 성능을 평가하기 위해 다음과 같은 실험 환경을 구성하였다. 실험에 사용된 구성요소와 컴퓨팅 사양은 표 5와 표6과 같다.

표 5. 실험 구성요소

구성요소	내용
IoT 디바이스	토픽 생성 및 센서를 통해 수집한 데이터와 디지털 서명 생성 후 브로커에 전달
브로커	IoT 디바이스로부터 수신한 데이터를 클라이언트로 전달 및 연결된 클라이언트에게 토픽 목록 전달
클라이언트	IoT 디바이스가 발행하는 토픽을 구독, 브로커로부터 수신한 디바이스의 데이터를 검증

표 6. 컴퓨팅 사양

운영체제	Windows11
CPU	Intel(R) Core(TM) i7-9700
RAM	32GB
GPU	NVIDIA GeForce GTX 2070

본 연구에서 무결성 검증 평가는 표 7에 정리한 네 가지 시나리오에 대해 실시하였다.

표 7. 무결성 검증 실험 결과

시나리오	검증 결과
1. (정상 상태) 원본 데이터의 변조가 없는 경우	성공
2. 가상환경에서 센서 원본 데이터가 변조된 경우	변조 감지
3. 브로커에서 원본 데이터가 변조된 경우	변조 감지
4. IoT 디바이스가 이탈 및 재접속된 경우	성공

시나리오 1은 데이터의 변경이 없는 경우로 시스템의 통신 시퀀스에 따라 IoT 디바이스에서 센서의 토픽으로 원본 데이터와 서명 데이터를 포함한 메시지를 발행하고 클라이언트의 가상환경은 브로커로부터 데이터의 변조 없이 메시지를 수신한다. 가상환경은 토픽에 포함된 DID로부터 연계된 공개 키를 블록체인에서 취득한 후 서명 데이터로 해당 메시지의 원본 데이터가 변조되지 않았음을 확인하였다.

시나리오 2는 가상환경에서 메시지를 수신한 후 원본 데이터를 의도적으로 변조한 후 원본 데이터의 무결성을 검증한다. 검증 알고리즘은 IoT 디바이스 지갑의 개인 키로 생성된 서명 데이터를 가상현실의 클라이언트에서 IoT 디바이스의 공개 키를 사용하여 수신한 원본 데이터의 무결성을 검증한다. 시나리오 2에서는 데이터가 변조되었음을 확인하였다.

시나리오 3은 브로커가 수신한 센서 데이터의 원본 데이터와 서명 데이터를 변조한 후 가상환경의 클라이언트로 송신하는 경우이다. 이는 시나리오 2와 같이 데이터의 변조가 네트워크 내에서 이루어지는 경우에도 원본 데이터가 변조되었음을 확인하였다.

시나리오 4는 IoT 디바이스가 시스템에서 이탈하고 다시 참가한 상황에서도 해당 IoT 디바이스의 무결성을 검증하는 데 문제가 없음을 확인하는 경우이다. 제안된 시스템에서 IoT 디바이스는 시스템에 참가할 때마다 새로운 DID와 공개 키를 생성하고 이를 블록체인에 저장하며 이에 따라 기존의 토픽 정보도 갱신된다. 따라서 가상환경은 브로커로부터 갱신된 토픽 정보를 받아와

재접속한 IoT 디바이스의 센서 데이터를 정상적으로 수신할 수 있어 디바이스가 시스템을 이탈한 후 다시 접속한 상황에서도 토픽을 갱신하여 무결성을 검증할 수 있음을 확인하였다.

다음으로 클라이언트와 디바이스 간 통신 지연시간을 비교하는 실험을 기존 MQTT 모델과 제안된 모델에 진행하여 통신의 실시간성을 평가한다. 이를 위해 클라이언트와 디바이스의 현재 시각을 동일하게 설정하였으며 지연시간은 (클라이언트의 디바이스 데이터 수신 시각) - (디바이스의 센서 데이터 측정 시각)으로 설정하였다. 실험은 5번 진행하였으며 실험 결과는 그림 9와 같다. x축은 실험 번호, y축은 지연시간을 의미한다.

실험 결과 기존 MQTT 모델과 제안된 모델의 지연시간 차이는 평균적으로 4.3ms로 나타났다. 이는 디지털 서명 생성과 서명 검증을 위해 블록체인에 접속하는 시간이 포함된 시간임을 알 수 있다.

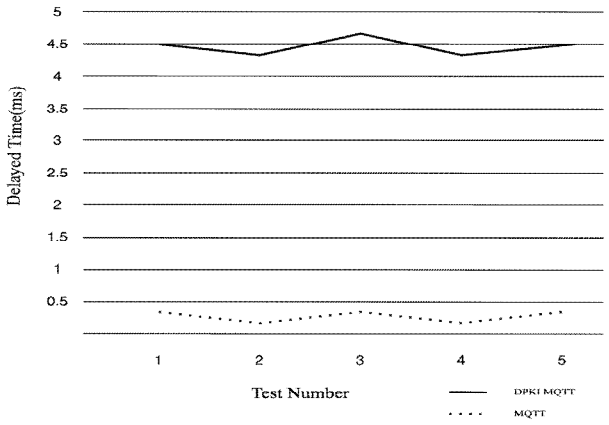

그림 9. 지연시간 비교 결과

V. 결 론

본 연구에서는 Hyperledger Indy의 DID 기술과 MQTT 프로토콜을 활용하여 메타버스와 같은 가상 환경에서 IoT 디바이스의 센서 데이터를 안전하게 전송하기 위해 설계 구현한 시스템의 요소 기술을 살펴보고 무결성을 검증하였다. 제안한 시스템은 MQTT 프로토콜의 토픽에 IoT 디바이스의 DID 정보를 포함하여 가상 환경에서 토픽 정보로부터 블록체인에 저장된 IoT 디바이스와 관련한 신뢰할 수 있는 공개 키, 암호화 알고리즘, 서비스 종단점 등을 취득할 수 있도록 하여 통신 주체 간 안전한 연결 및 메시지의 신뢰성을 보장한다.

향후 제안 시스템을 확장된 메타버스에 적용하면서 발생할 수 있는 문제점과 무결성을 보장하고 MQTT 브로커를 탈중앙화하는 방안 및 복수의 클라이언트가 하나의 토픽을 구독할 때 통신 채널을 안전하게 구현하는 방법을 연구할 계획이다.

REFERENCES

[1] 김학서, 김용운, 선경재, 유상근, “메타버스 서비스를 위한 디지털 트윈 기반 현실 -가상 융합 인터랙션 기술 동향,” *한국통신학회지*, 제40권, 제11호, 24-31쪽, 2023년

[2] B. Ko and M. Kim, “Metaverse-based immersive content R&D supportbusiness trend,” *Broadcasting and Media Magazine*, vol. 27, no. 1, pp. 21-26, 2022.

[3] 이영우, 권혜민, 문한솔, 최현범, 이혜민, 서정욱, 이창교, “IoT 환경 센서와 메타버스의 디지털 트윈 간 실시간 연동 구현,” *한국통신학회 학술대회논문집,* 325-326쪽, 2022년 6월

[4] What is MQTT?, https://aws.amazon.com/ko/what-is/mqtt/, (accessed Dec., 02, 2023).

[5] N. Rodday et al., "The Resource Public Key Infrastructure (RPKI): A Survey on Measurements and Future Prospects," in *IEEE Transactions on Network and Service Management,* Oct. 2023.

[6] N. H. Kim and C. S. Hong, “Secure MQTT Protocol based on Attribute-Based Encryption Scheme,” *.J. of KIISE,* vol. 45, no. 3, pp. 195-199, 2018.

[7] 최규현, 김근형, “자기주권 신원 생태계를 위한 신뢰할 수 있는 통신 방법,” *한국정보처리학회,* 제11권, 제3호, 91-98쪽, 2022년

[8] 홍유성, 김근형, “신뢰할 수 있는 탈중앙 메타버스를 위한 IoT 통신 기술 개발,” *한국디지털콘텐츠학회 학술대회논문집,* 157-159쪽, 2023년 11월

[9] DID Implementation Guide v1.0(2021), https://www.w3.org/TR/did-imp-guide/, (accessed Dec., 04, 2023).

[10] A. Papageorgiou, A. Mygiakis, K. Loupos and T. Krousarlis, "DPKI: A Blockchain-Based Decentralized Public Key Infrastructure System," *2020 Global Internet of Things Summit (GIoTS), Dublin, Ireland, 2020,* pp. 1-5, 2020.
[11] F. Buccafurri, V. D. Angelis and R. Nardone, "Securing MQTTT by Blockchain-Based OTP Authentication," *Sensors 2020,* vol. 20, no. 7, 2002.
[12] M. A. Abubakar, Z. Jarocheh, A. and Al-Dubai, X. Liu, "Blockchain-based identity and authentication scheime for MQTT protocol," *ICBCT' 21, ACM,* pp. 73-81, Mar. 2021.
[13] A. Dixit, M. S.-Creasey and M. Rajarajan, "A Decentrqalized IIOT Identity Framework based on Self-Soverign Identity using Blockchain," *IEEE LCN 2022,* pp. 3350338, Sep. 2022.
[14] N. Fotiou, V. A. Siris, G. C. Polyzos, Y. Kortesniemi, D. Lagutin, "Capabilties-based access control for IoT device using Verifiable Credentials," *SPW 2022*, pp. 222-228, May 2022.
[15] 남혜민, "사물인터넷 환경에서 발행 구독 프로토콜에 적용가능한 분산식별자 기반의 인증기법 연구", *단국대학교 석사학위 논문*, 2021년
[16] A. Philipp, A. Kupper, "DAXiot: A Decentralized Authentication and Authorization Scheme for Dynamic IoT Network," arXiv: 2307.06919v2.
[17] GitHub 저장소, https://github.com/Hongyoosung/Metaverse_for_IoT(accessed Dec., 06, 2023).
[18] M2MqttUnity(2017), https://github.com/gpvigano/M2MqttUnity, (accessed Dec., 06, 2023).
[19] hyperledger/indy-sdk, https://github.com/hyperledger/indy-sdk, (accessed Dec., 06, 2023)
[20] MQTTnet, https://github.com/dotnet/MQTTnet, (accessed Dec., 06, 2023).
[21] paho-mqtt 1.6.1(2021), https://pypi.org/project/paho-mqtt/, (accessed Dec., 08, 2023).

저 자 소 개

홍유성(학생회원)

2019년 연초고등학교 졸업.
2024년 ~ 현재 : 동의대학교 게임공학과 재학.
<주관심분야 : 가상환경, 인공지능, 블록체인>

김근형(정회원)

1986년 서강대학교 전자공학과 학사 졸업.
1988년 서강대학교 전자공학과 석사 졸업.
2005년 포항공과대학교 컴퓨터공학과 박사 졸업.
2007년 ~ 현재: 동의대학교 게임공학과 교수
<주관심분야 : 탈중앙웹, 블록체인, 자기주권 데이터, 인공지능, 설명가능 인공지능>