J Inf Process Syst, Vol.19, No.1, pp.89~97, February 2023
https://doi.org/10.3745/JIPS.01.0093

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Design and Implementation of a Blockchain Based Interworking of oneM2M and LWM2M IoT Systems

Donggyu Kim[1], Uk Jo[1], Yohan Kim[1], Yustus Eko Oktian[2], and Howon Kim[1,*]

Abstract

With the growth of Internet-of-Things (IoT) technologies, the number of IoT devices developers need to manage has increased exponentially. Many IoT standards have been proposed to allow those devices to communicate efficiently in day-to-day tasks. However, we lack trusted interworking entities for devices from different standards to collaborate securely. This paper proposes a blockchain platform that bridges multiple heterogeneous IoT platforms to co-exist and interwork. Specifically, we designed an interworking proxy application entity (IPE) implemented as a chaincode in Hyperledger Fabric to collect and process data coming from/to oneM2M and LWM2M architecture. The use of blockchain will guarantee network reliability and data integrity so that cross-standard communications can be audited and processed securely. Based on our evaluation, we show that the interworking between oneM2M and LWM2M through our blockchain platform is feasible. Furthermore, the proposed system can process up to 206 transactions per second with 1,000 running applications, which is about an 87% increase from the previously referenced study.

Keywords

Blockchain, IPE (Interworking Proxy application Entity), IoT Platforms

1. Introduction

Internet-of-Things (IoT) technology was initially used only in industrial fields, but recently, they are used in a wide range of businesses and day-to-day lives, such as home appliances, smartphones, and smartwatches [1]. As a result, the number of IoT devices and the data that needs to be processed increase exponentially. Several IoT standard platforms have been proposed to manage IoT data efficiently, but data transmission or device interworking between different platforms may not work correctly due to interoperability problems [2]. Therefore, the role of interworking proxy application entity (IPE) that provides interoperability between different platforms for seamless communication between heterogeneous IoT platforms is becoming more important. In addition, current IoT data is being collected and managed from a centralized private server or cloud server, emphasizing practicality. However, this method generally has a single point of attack that is easy for a malicious user to attack. Potentially, an attacker can compromise the stored data, making the data unreliable for users.

Manuscript received August 8, 2022; first revision October 14, 2022; accepted November 27, 2022.

*** Corresponding Author:** Howon Kim (howonkim@pusan.ac.kr)
[1] Dept. of Computer Engineering, Pusan National University, Busan, Korea (donggyu@islab.re.kr, jouk@islab.re.kr, yohan@islab.re.kr, howonkim@pusan.ac.kr)
[2] Blockchain Platform Research Center, Pusan National University, Busan, Korea (yustus@islab.re.kr)
Donggyu Kim and Uk jo contributed equally to this work.

This paper is an extended version from our paper previously published in ASK 2022 academic conference of KIPS.

This paper proposes a novel heterogeneous IoT platform interworking platform using Hyperledger Fabric blockchain to solve the previously mentioned interworking and data centralization problem. Using the features of blockchain, IPE implemented as smart contracts can provide a deterministic and verifiable standard for IoT interoperability through the smart contract and provide a secure decentralized peer-to-peer architecture through the blockchain consensus. The proposed platform uses events occurring on the blockchain to transmit data to diverse IoT platforms connected via blockchain nodes. This data is stored in the blockchain and shared by all participating blockchain nodes. Furthermore, the privacy level of the shared data can also be configured via channel access permissions.

Technically, this paper highlights the interoperability scenarios between oneM2M and LWM2M as they are one of the most used IoT platforms in the market. However, our architecture design is generic such that other IoT platforms can also be added in the future when the interfaces are implemented. In summary, we made the following contributions:

- We redesign the IPE to be workable in the blockchain environment to support interoperability between oneM2M and LWM2M architecture.
- We provide proof of concept for our IPE design, which is implemented as a chaincode in the Hyperledger Fabric network.
- We evaluate our implementation to analyze its feasibility, performance, and scalability.

The rest of this paper is organized as follows. We first discuss the background of the IoT platform and the existing heterogeneous IoT interworking model in Section 2. We then describe details of our proposed model in Section 3, while its evaluation is presented in Section 4. Finally, we provide literature reviews on the existing works in Section 5 and conclude in Section 6.

2. Preliminaries

This section describes terms and related technologies that are used in our paper.

2.1 oneM2M

The oneM2M standard is global technical standard for interoperability for machine-to-machine and IoT technologies based on requirements contributed by its members. The standard consists of a three-layer model according to its function, and in this paper, we use the application entity (AE), which provides IoT service application logic, and common service entity (CSE), which provides common service functions of oneM2M platforms. These entities may communicate through an interface called a reference point. For example, CSE provides common service capabilities to CSEs in AE or CSE of other domains.

2.2 LWM2M

Lightweight machine-to-machine (LWM2M) is one of the protocols for IoT device management and service enablement. LWM2M is based on the constrained application protocol (CoAP), and it is suitable for use by devices with slow transmission rates. Specifically, the LWM2M standard defines the functions and interfaces between an LWM2M server and an LWM2M client located in an IoT device. Access to the IoT resources in the IoT devices is done through a URL. Because the network resources are managed efficiently, LWM2M servers can connect and control a large number of IoT devices.

2.3 Interworking Proxy Application Entity

Since oneM2M and LWM2M are different protocols, an interworking process is required to enable both protocols to work together. OneM2M standard defines IPE as an interface that interoperates with external systems in CSE for interworking with other platform systems. LWM2M can use this interface to create LWM2M IPE that will bridge LWM2M applications to CSE.

2.4 Hyperledger Fabric

Hyperledger Fabric is a permissioned blockchain framework that can be tweaked to meet several corporate use cases. Each organization can participate in the blockchain network by deploying a peer (a node within the organization that has a ledger and chaincode) and an orderer that controls the order of transactions to be recorded in the blockchain. Furthermore, organizations can also create private channels within the blockchain network to process business logic and share data among related organizations.

3. System Model

3.1 System Architecture

Fig. 1 represents the overall design of the platform proposed in this paper. The oneM2M platform and LWM2M platform interwork to transmit data between one another. The blockchain is used to prevent data collected from sensors or IoT devices from being stored in a centralized structure. The blockchain used in the platform is Hyperledger Fabric, a permissioned blockchain that aims to be applied to a private IoT network. There are three major components of the platform: Hyperledger Fabric network, Fabric Bridge gateway, and external client application.

A Hyperledger Fabric network consists of channels, peers, and orderers. A smart contract called chaincode is installed in all peers constituting the Fabric network, and IoT data is stored in the blockchain through peer consensus process through chaincode. The name of the chaincode is IPE, and it is written in Go language and is divided into three layers: Contract, Model, and Service. OneM2M data and LWM2M data models are defined in the chaincode, and data required for platform interworking is extracted from this model and transmitted together with routing information. The chaincode can also generate events that clients connected to each peer can receive on the blockchain. The chaincode event contains information about the data the sender wants to send and the client that will receive the data.

Fabric Bridge gateway has Fabric SDK and function modules that use it. In Fabric Bridge gateway, oneM2M platform and LWM2M platform are virtualized and executed in the form of containers for data transmission and reception between IoT devices and gateways.

External applications have the Fabric SDK and the IoT platform client they use. For communication with the blockchain platform, a request can be sent directly to a server connected to the Fabric Bridge gateway or to the Hyperledger Fabric network, and events generated by the blockchain platform can be received and data can be processed.

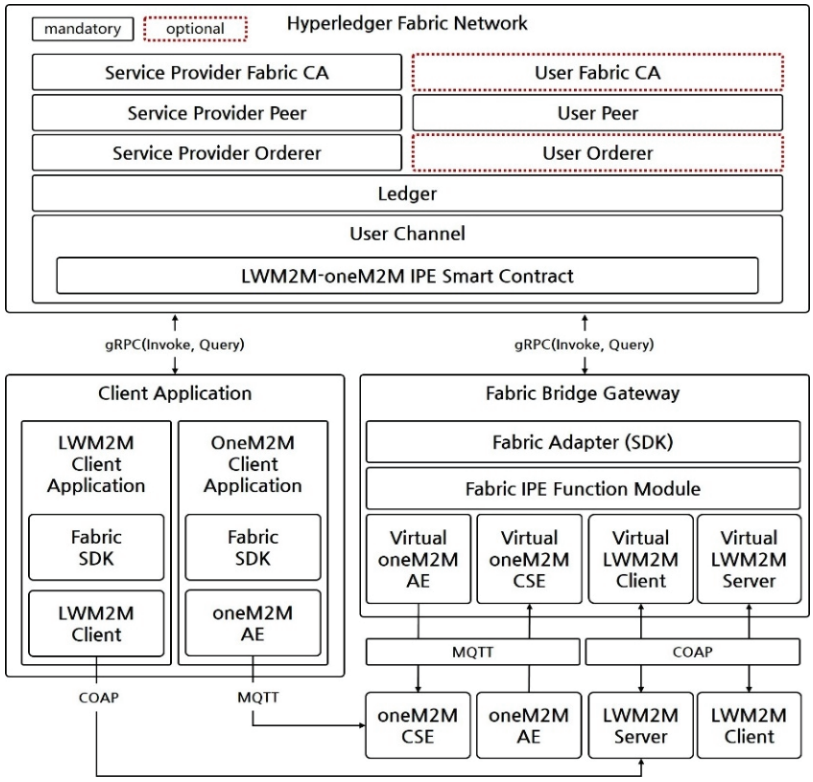


Fig. 1. Overall system architecture from our proposal, which enables interworking between oneM2M and LWM2M through Hyperledger Fabric network.

3.2 Interworking Process

When a block containing a request transaction is created in a blockchain network, an event is generated and sent to all peers in the network. When a peer in the network receives an event, it checks the destination routing information and sends the data to the bridge gateway or client application that matches the destination routing information. In addition, our platform implements two-way data interworking from LWM2M to oneM2M and from oneM2M to LWM2M. The specific operation sequence of the proposed platform is as follows.

Fig. 2 shows how the oneM2M platform and the LWM2M platform exchange data with each other. Two operation scenarios are shown in the sequence diagram above. The first operation scenario shows a process in which oneM2M application transmits oneM2M platform data to LWM2M server. The second operation scenario shows the process in which the LWM2M client transmits data to the oneM2M application through the LWM2M protocol.

The first step in the work process before starting the scenario is to install the necessary services on the IPE-Fabric Bridge gateway: oneM2M platform to send and receive oneM2M data, and LWM2M platform to send and receive LWM2M data. And we start Scenario 1 after setting up the blockchain and gateway by installing the chaincode on the blockchain.

In the first scenario, oneM2M application registers itself as a fabric event listener (1-1). Then it calls the chaincode to send the request (1-2). The Hyperledger Fabric network validates the chaincode (2-2)

and sends the result to the IPE-Fabric Bridge gateway through an event and stores it in the Ledger (2-3). IPE-Fabric Bridge gateway checks the event (2-4). This event contains the data sent by the application and routing information about where to send the data. The IPE-Fabric Bridge gateway sends the data transmitted as an event to the LWM2M server corresponding to the routing information (2-5). LWM2M server processes the request and sends a response to the IPE-Fabric Bridge gateway (2-6). The IPE-Fabric Bridge gateway calls the chaincode to send the received response to the Hyperledger Fabric network (2-7). The Hyperledger Fabric network validates the chaincode (2-8) and sends the result to the oneM2M application through an event and stores it on the Ledger (2-9).

In the second scenario, the LWM2M client sends a request to the IPE-Fabric Bridge gateway (3-1). Upon receiving the client's request, the IPE-Fabric Bridge gateway calls the chaincode to send the request to the Hyperledger Fabric network (3-2). The Hyperledger Fabric network validates the chaincode (3-3) and sends the result to the oneM2M Application through an event and stores it in the Ledger (3-4). oneM2M application checks the event (3-5) and sends a response by calling the chaincode (3-6). The Hyperledger Fabric network validates the chaincode (3-7) and sends the result to the IPE-Fabric Bridge gateway through an event and stores it in the Ledger (3-8). The IPE-Fabric Bridge gateway transmits the response received as an event to the LWM2M client (3-9).

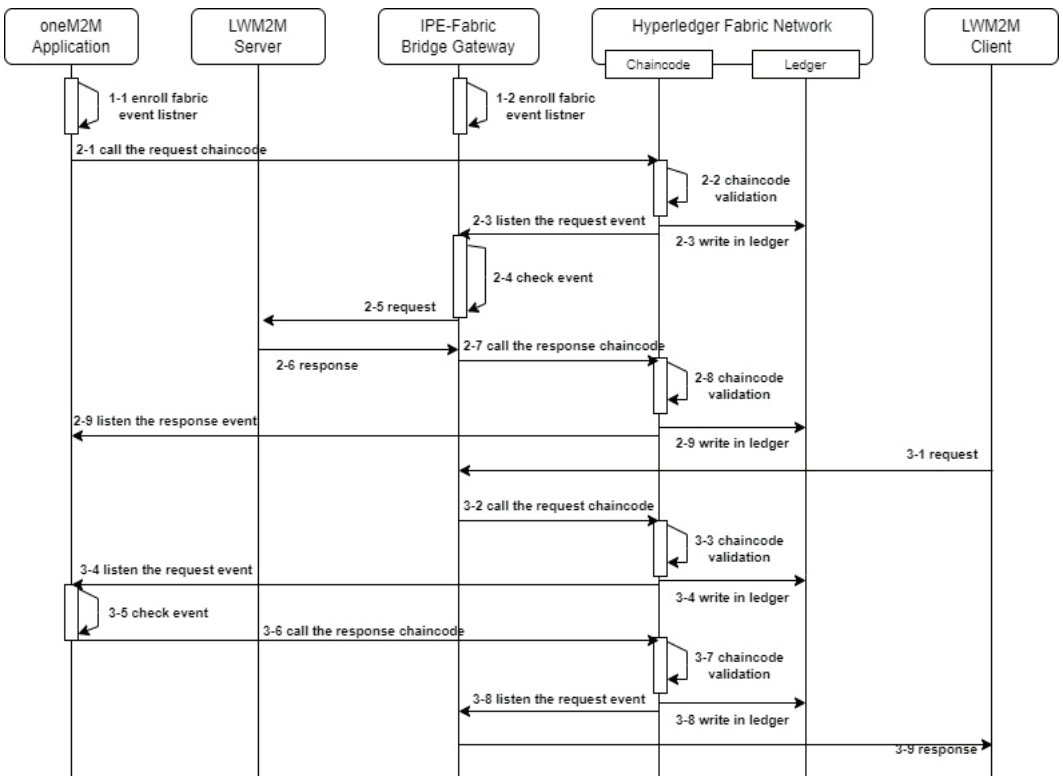


Fig. 2. Data transfer between oneM2M and LWM2M based on Hyperledger Fabric chaincode events. In the first scenario, the oneM2M application wants to send data to the LWM2M server from (2-1) to (2-9). In the second scenario, the LWM2M client wants to send data to oneM2M application through the IPE-Fabric Bridge gateway from (3-1) to (3-9).

The advantages obtained through the above operation are as follows.

- Since LWM2M platform data and oneM2M platform data can be linked with each other, the proposed blockchain platform can act as an IPE. In addition, all peers participating in the blockchain have the same data, which increases the data availability.
- Data can be prevented from being centrally stored because it uses a permissioned blockchain to replicate and store data to all peers participating in the blockchain. In addition, it is very difficult to arbitrarily manipulate data due to the nature of hard-to-tamper guarantee from the blockchain, which increases the reliability of data and provides security advantages.

4. Experimental Results

The application connected to blockchain peer is implemented as the node.js application in a PC (Ubuntu 18.04, Intel i7-10700k 3.8 GHz, 16 core, 64 GB RAM). We then build a blockchain network using Hyperledger Fabric 2.2 in same PC. The blockchain was deployed with 2 peers and 3 orderers and 1 channel. The ordering service runs Raft consensus algorithm.

We evaluate our proposal in terms the performance throughput and scalability. The throughput is measured as the number of transactions per second (TPS) that our proposal can handle with respect to the number of applications (apps) available in the blockchain network. Specifically, we vary the number of applications connected to the blockchain peers from 1,000 to 2,000. They will transmit data to our Hyperledger Fabric network, and then we measure the TPS using Apache JMeter. The results are shown in Fig. 3.

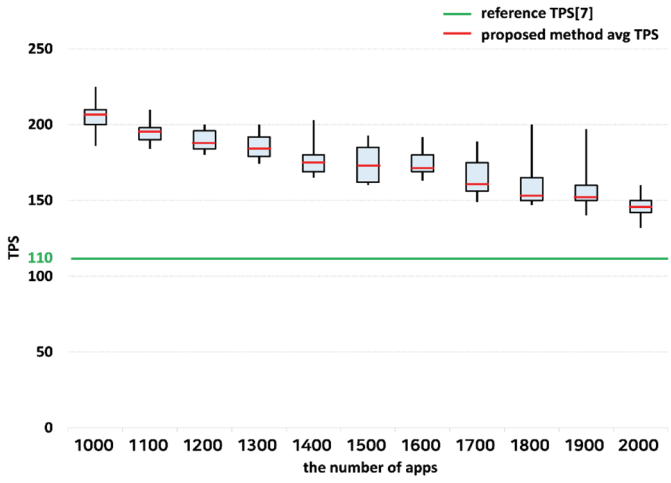


Fig. 3. Throughput of the proposed system in transaction per second (TPS) based on the number of applications connected to the blockchain peer.

The TPS throughput decreases linearly as we increase the number of connected apps. For example, when 1,000 applications are connected, we confirm that the system can process up to 206 TPS. Meanwhile, it declines to 148 TPS when 2,000 applications are present. This behavior is expected because the system handles more workloads when more applications exist, and this trend is common in other system benchmarking. On average, our system can process about 160 TPS, which is better than the 110 TPS previously measured in [3].

However, our throughput still shows a lower performance when compared to conventional centralized databases. To improve the performance further, we can use a Hyperledger blockchain acceleration tool such as Samsung's Nexledger Accelerator [4]. This external accelerator receives and classifies transactions from clients on behalf of blockchain nodes, which can further improve the original TPS by up to 15 times.

5. Related Work

Several blockchain-based IoT data management and interworking between IoT protocols have been proposed in the literature. Lee et al. [5] try to strengthen the confidentiality of data by using Logchain to manage IoT data access rights. The proposed Logchain is designed not to affect the performance of existing IoT systems by limiting the use case only to manage IoT access. However, the proposed method does not have interworking capabilities to interchange data to/from different domains, such as in IPE.

Similar to our paper, Kim et al. [6] study the interworking between heterogeneous platforms. They present a model capable of data transmission from LWM2M to oneM2M. However, their approach is only one-directional. The data transmission mechanism from oneM2M to LWM2M is not supported. Furthermore, they only propose a design without actual implementation. In contrast, we provide bi-directional interworking between oneM2M and LWM2M with a proof-of-concept implementation using Hyperledger Fabric. Liu et al. [3] designed a blockchain network for IoT data access on Hyperledger Fabric. The proposed study shows that adaptive block size plays an essential role in performance improvement. Furthermore, maximum performance can be achieved if other scalability solutions are considered. The table below provides feature comparisons of our paper with other papers.

The use case of the papers compared in Table 1 is limited to data logging or a specific service, whereas the use case of our platform is not limited to a specific scope, and there is a difference in that it constitutes a general IoT workflow [5,7-10]. As for the supported IoT platform, there are papers that support more than two platforms, but our platform is the only platform that supports interworking using blockchain.

Table 1. Feature comparison between our paper and related studies

Study	Use case	IoT platform			Blockchain platform	Support inter-working (IPE)
		oneM2M	LWM2M	OCF		
Witanto et al. [7]	IoT firmware update	No	No	Yes	Ethereum	No
Jeong et al. [8]	Intelligent healthcare	Yes	Yes	Yes	Not specified	No
Martin et al. [9]	IoT web application	No	Yes	No	Ethereum	No
Lee et al. [5]	IoT data logging	Yes	No	No	Hyperledger	No
Lee et al. [10]	IoT data logging	Yes	No	No	Private blockchain	No
Proposed	General IoT workflow	Yes	Yes	No	Hyperledger	Yes

6. Conclusion

This paper proposed a blockchain-based platform to support interworking between multiple different IoT platforms. We designed an IPE re-implemented as a chaincode in Hyperledger Fabric to collect and

process data coming from/to oneM2M and LWM2M architecture. Since it is based on a blockchain, all peers can see the same data, increasing data usability and solving reliability and security issues that previously existed in typical non-blockchain systems. Our experimental results show that the proposed system can process up to 206 transactions per second with 1,000 running applications, which is about an 87% increase from the previously referenced study. For future work, we will investigate the integration of our proposal with blockchain acceleration tools such as Samsung's Nexledger Accelerator to improve the performance and scalability further.

Acknowledgement

This work was supported by a 2-Year Research Grant of Pusan National University.

References

[1] K. A. Ogudo, D. Muwawa Jean Nestor, O. Ibrahim Khalaf, and H. Daei Kasmaei, "A device performance and data analytics concept for smartphones' IoT services and machine-type communication in cellular networks," *Symmetry*, vol. 11, no. 4, article no. 593, 2019. https://doi.org/10.3390/sym11040593

[2] J. Yun, S. C. Choi, N. M. Sung, and J. Kim, "Towards global interworking of IoT systems: oneM2M interworking proxy entities," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, Seoul, South Korea, 2015, pp. 473-474.

[3] C. M. Liu, M. Badigineni, and S. W. Lu, "Adaptive Blocksize for IoT payload data on fabric blockchain," in *Proceedings of 2021 30th Wireless and Optical Communications Conference (WOCC)*, Taipei, Taiwan, 2021, pp. 92-96.

[4] K. H. Kwak, J. T. Kong, S. I. Cho, H. T. Phuong, and G. Y. Gim, "A study on the design of efficient private blockchain," in *Computational Science/Intelligence & Applied Informatics*. Cham, Germany: Springer, 2019, pp. 93-121.

[5] C. Lee, L. Nkenyereye, N. Sung, and J. Song, "Towards a Blockchain-enabled IoT platform using oneM2M standards," in *Proceedings of 2018 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, South Korea, 2018, pp. 97-102.

[6] D. C. Kim, Y. H. Kim, Y. Kwon, and H. W. Kim, "Proposal of IoT platform interworking framework model using blockchain technology," *Proceedings of the Korea Information Processing Society Conference*, vol. 29, no. 1, pp. 124-127, 2022.

[7] E. N. Witanto, Y. E. Oktian, S. G. Lee, and J. H. Lee, "A blockchain-based OCF firmware update for IoT devices," *Applied Sciences*, vol. 10, no. 19, article no. 6744, 2020. https://doi.org/10.3390/app10196744

[8] S. Jeong, J. H. Shen, and B. Ahn, "A study on smart healthcare monitoring using IoT based on blockchain," *Wireless Communications and Mobile Computing*, vol. 2021, article no. 9932091, 2021. https://doi.org/10.1155/2021/9932091

[9] C. Martin, I. Alba, J. Trillo, E. Soler, B. Rubio, and M. Diaz, "Providing reliability and auditability to the IoT LwM2M protocol through blockchain," 2020 [Online]. Available: https://arxiv.org/abs/2008.06694.

[10] C. Lee, N. M. Sung, L. Nkenyereye, and J. Song, "Demo Abstract: Blockchain enabled Internet-of-Things Service Platform for Industrial Domain (No. 595)," 2018 [Online]. Available: https://easychair.org/publications/preprint/5rlb

Donggyu Kim **https://orcid.org/0000-0002-9151-7202**

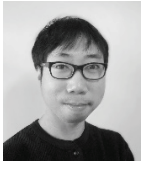
He received B.S. degree in Computer Science Engineering from Pusan National University in 2021. Since September 2021, he is with the School of Computer Science Engineering from Pusan National University as a M.S. candidate. His current research interests include blockchain.

Uk Jo **https://orcid.org/0000-0002-2332-8573**

He received B.S. degree in electrical engineering from Kwangwoon University in 2012, M.S. degree in electrical engineering from Kwangwoon University in 2016. Since March 2022, he is with the School of Computer Engineering from Pusan National University as a Ph.D. candidate. His current research interests include blockchain and security.

Yohan Kim **https://orcid.org/0000-0002-2892-4649**

He received B.S. degree in Computer Science Engineering from Pusan National University in 2022. Since March 2022, he is with the School of Computer Science Engineering from Pusan National University as a M.S. candidate. His current research interests include blockchain.



Yustus Eko Oktian **https://orcid.org/0000-0002-3574-7820**

He is currently a postdoctoral researcher at Pusan National University, South Korea. He received his bachelor's degree in Electrical Engineering from Petra Christian University, Indonesia, in 2013, and his master's and doctoral's degree in Computer Engineering from Dongseo University, South Korea, in 2016 and 2021. His research interests are network security, distributed computing, blockchain, Internet of Things (IoT), and software-defined networking (SDN).

Howon Kim **https://orcid.org/0000-0001-8475-7294**

He is currently a professor at the Department of Computer Engineering, Pusan National University and chief of Energy IoT Research Center and Chief of Blockchain Platform Research Center. Before joining Pusan National University in 2008, he worked at the Electronics and Telecommunications Research Institute (ETRI) since 1998. He visited the Chair for the Communication Security Group (COSY) in Ruhr-University Bochum, Germany, as a postdoctoral researcher from July 2003 to June 2004. He received a Ph.D. from Pohang University of Science and Technology (POSTECH) in 1999. His research interests are blockchain platform, cryptography, deep learning, and security chip design.