Blockchain-based Multi-Purpose Authentication Method for Anonymity and Privacy

Yong Joo Lee, Keon Myung Lee

Abstract: ***Various applications using smart contract, a leading application technology of blockchain, are being rapidly introduced to the industrial sector. As a result, services in various fields are actively being developed. Currently, most of the services are offered on a variety of platforms, not blockchain-based. If these services are linked to prepaid features that provide anonymity in smart contracts, a more diverse service scenario could be created. In this paper, we propose scenarios that provide certification for various purposes based on smart contracts. It provides a scenario that provides the privacy of the contract signed by the customer while retaining the anonymity provided by blockchain. Smart contracts register keys that do not give a clue to guess the encoding keys and deliver hash functions of the child keys that change each time with authentication parameters.In addition, the master seed that can generate these authentication parameters is designed to be kept only by the user and the service provider to be able to verify them. It is proposed by considering both a single service provider transaction and a smart contract authentication model that is shared with a large number of service providers. To generate these child keys, we proposed a mechanism to use the method of generation of child keys based on the Elastic Curve Cryptography public-key method. Various attack scenarios were analyzed to complement the scenario and the efficiency of the proposed mechanism was analyzed. In addition, differences and excellence were compared by organizing scenarios that had the same purpose as scenarios in the relevant study.***

Keywords :Blockchain, Authentication, Role-based Access Control, Anonymity, Privacy.

I. INTRODUCTION

In a centralized network architecture of an existing client and server architecture, All responsibilities, including service delivery and security, were concentrated on the central server.Due to the spread of IoT applications and the development of various sensor technologies, the responsibility for the ever-increasing lack of storage capacity, bandwidth and power for a huge number of objects has increased [1, 2, 3].Thus, a paradigm shift to a decentralized network structure is presented as a solution that can address this.Blockchain technology, which is leading the shift to peer-to-peer architecture, characterized by all nodes playing an equal role and without a single point of failure, makes it impossible to tamper with information registered in distributed principals, thereby making the "trustless network" a reality.It is also leading the de-centralization drive by giving users the authority to "Automatic Control" to manage their own security [4, 5].Among these blockchain technologies, we would like to propose authentication scenarios that provide anonymity with existing service providers by utilizing smart contractcurrently applied to a variety of industries.In chapter 2, we analyze the technologies and studies involved, in chapter 3 we details the authentication techniques proposed, and try to verify and conclude thetechniques proposed in terms of security and efficiency.

II. PROCEDURE FOR PAPER SUBMISSION

A. Related Works

The authentication algorithm proposed in this paper is based on the ECDSA method.ECDSA-based disposable open-key algorithms are introduced first, followed by two related studies.The first related study is a scenario in which blockchain-based authentication is provided for the provision of existing services rather than blockchain-based, and the second related study is a scenario used in conjunction with the authentication protocol to implement RBAC using smart contact.

B. SCDSA Algorithm

BIP32 is currently a Child Public Key Generation algorithm used in the electronic wallet of Bitcoin and is based on the Elliptic Curve Cryptography (ECC) method.The ECC approach has the advantage of providing high security compared to the RSA, as public key pairs are created based on the EC domain parameters consisting of six parameters. Currently, in the electronic wallet used in Bitcoin, the child public key is generated and used as the address of the branched account, but the algorithm proposed in this paper is intended to utilize the method of creating the child secret key to apply to the requirement, triangle authentication algorithm. The method of creating a secret key is as shown in Equation (1, 2).$\hat{Q}$is the master public key and $\hat{d}$ is the master secret key.Q_i Child Public Key and d_i is Child Public Key.$j$is a random variable that is generated randomly every time [6, 7].

Revised Manuscript Received on July 22, 2019.

YongJoo Lee, Department of Computer Science, Chungbuk National University, Cheongju, Chungbuk 361-763, Korea, silvianna817@naver.com

Keon Myung Lee, Department of Computer Science, Chungbuk National University, Cheongju, Chungbuk 361-763, Korea,

$$Q_j = \hat{Q} + (hash(j, \hat{Q})(modp) \quad (1)$$

$$d_j = \hat{d} + (hash(j, \hat{Q})(modp) \quad (2)$$

C. Blockchain-based Authentication

Mobile users have proposed scenarios for authentication with various service providers without membership [3]. In this algorithm, blockchain is utilized in the authentication scenario of service provider that has blockchain channel but provides various services that do not use blockchain channel.The user has used Virtual ID and Public key as authentication parameters for service delivery without membership, but after authentication, the user information is released to the public, which results in the loss of the purpose of the user.Failure to protect user information may result in threats such as disclosure of information.Encryption that was used without considering the mobile environment, which is a user terminal, is also not clearly intended for use [8, 9].

Implementing too much encryption and decryption without a specific purpose in a mobile environment will result in a lot of traffic, resulting in less efficiency and unnecessary processes.By placing the blockchain in an intermediate role with the server, it was not possible to realize the trustless, the greatest advantage of blockchain, by using it only as a trusted broker.In addition, users do not have the effect of using blockchain as they rely on centralized servers for all personal information.Users should be able to control themselves and trade without a trusted third party without relying on servers, but they are not.

The algorithm of service providers proposed in this paper is an authentication scenario that is currently available to service providers that provide a variety of services rather than blockchain-based.The service provider will be able to extract the user's phone number after authentication. It will also request a late payment to the mobile server based on the number extracted. This is the advantage that mobile servers generate revenue without operating services.The prepaid function using blockchain is the biggest advantage in terms of service providers in the anonymous payment system, but it does not take advantage of that advantage and introduced the existing method. As a result, only the system becomes more complex and has no merit. The figure 1 illustrates this approach.Figure 1 shows the flow between users, servers and blockchains,and service providers.

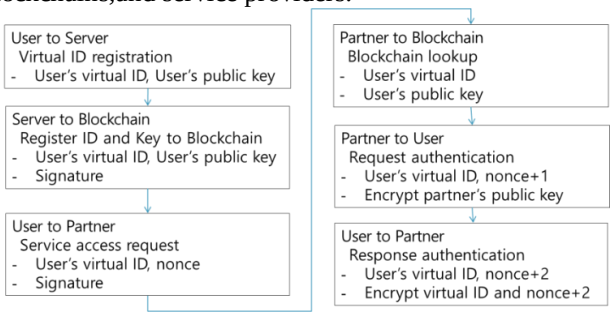

Fig 1. Scenario for Blockchain-based authentication for mobile service

D. Role-based Access Control-Smart Contract

J.CRUZ[7] proposed a role-based authentication method using smart apps. They have designed steps to define the relationship between users, roles, and services, assign roles that are appropriate for all users, and validate roles to use specific services. The role was registered using Smart Contract to provide transparency in the role, and the Challenge-Response protocol was defined to prevent user-disguise attacks.Implemented and released using Ethereum and Solidity, the purpose of this study is to validate the role given to users.By having the role issuer register for the role's transparency, users will not be able to register their own roles directly, eliminating the connection between the role and the account.In addition, even though the role was registered, the account could not be authenticated through it, resulting in redundancy of the authentication protocol called the challenge and response protocol [10, 11].Because channels using smart contact and authentication channels were duplicated, smart contact was only used as a repository for roles.Since users cannot post their own information on their own, they are not able to take advantage of the greatest strengths because they are difficult to manage their information.There are disadvantages of not connecting users to smart contact channels and thus not being able to connect to payment functions of various smart devices.Figure 2 shows a role-based service scenario.

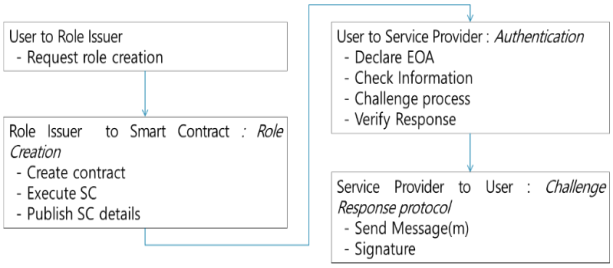

Fig 2. Scenario for RBAC for Smart Contract

III. PROPOSED MECHANISM

In the paper, the authentication algorithm available for various purposes in blockchain is proposed.The authentication algorithm is used as a method to provide anonymity using ECC-based disposable key algorithms and is designed to be applied for various purposes, including various payment information and role information. If a key is shared with one service provider in consideration of efficiency and safety, the key was encrypted with the service provider's public key to increase safety and omit the process for threats.On the other hand, if a key is shared with multiple service providers, the encryption was omitted to eliminate the key distribution procedure and the parameters were inserted to eliminate the various threats. To provide the integrity of these parameters, digital signatures are added so that more diverse information can be provided at once.

A. Environment of the proposed mechanism

The user authentication method proposed in this paper is a certification for various purpose

that provides a variety of purposes.The first function that the proposed mechanism provides is to provide authentication for payment by connecting smart contact channels with service providers that provide a variety of services.The service user creates the contract using the provider and smart contact channel, registers the authentication key with the smart contact channel and requests authentication to receive the service provider's service through various channels.The service provider completes the payment by providing the service and submitting the predefined terms.The second purpose of the offering is to provide role-based access control.Among the following figures 3, (a) shows the certification environment for service delivery in various environments for the first purpose, while (b) shows the environment using the same mechanism for role-based access control.

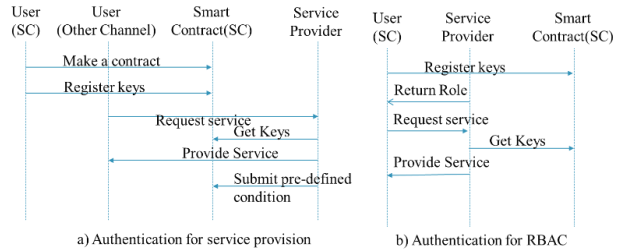

Fig 3. Two scenarios for the proposed authentication mechanism

B. Authentication Procedures

The protocol proposed in this paper generates authentication keys using BIP32Wallet, which is a method of generating child keys based on ECC public key method [10, 11].In the ECC method, $\hat{d}$ plays a role in generating keys for encryption and extracting the decoding keys for encoding, so it serves as a secret key to keep them secret.However, the scenario proposed in this paper is not en/decoding, but attempts to authenticate simply by creating a child key, so there is no distinction between the secret key and the public key's role [8, 9].The generated child key is transferred by changing to a hash value that is used only once to prepare for various network attacks. The proposed authentication method consists of three stages: registration, request, and performance.Smart Contract reveal keys that do not provide a clue to guess other keys even after being released, and only the user keeps the master seed that can generate the child key in secret.In addition, the values for verification were transferred to the single-use hash function to enable the service provider to verify only [12].In the following figure 4, (a) is the default scenario for sharing authentication keys with multiple service providers, and (b) is for use only with smart contact channels with specific service providers and is registered with the service provider's public key encryption.This allows only service providers with the appropriate secret key to view, thus providing more secure authentication parameters.The reason why the first scenario did not register securely with the secret key is that encryption of keys that should be shared with multiple service providers would complicate the distribution of decoding keys.Therefore, the key should be disclosed and included a method to analyze and block the type of attack that could result from it.Scenario (b) only allows service providers to decode keys by removing the method of blocking attacks and encrypting them with the service provider's disclosure key.

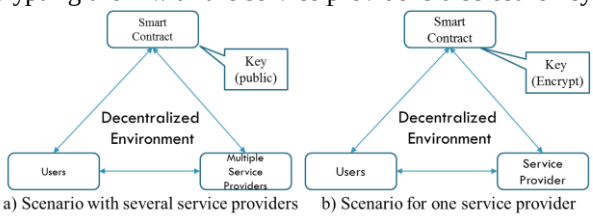

Fig 4. Design rule for the two scenarios of the proposed authentication mechanism

The procedure is about registering the authentication key on the provided channel after assuming that the smart contact channel with the user and service provider is ready.Assume that all keys and parameters generated by the proposed protocol are valid.The user creates an EC domain parameter $T = (p, a, b, G, n, h)$ as shown in (2), selects $d_i^{\wedge} \in [1, n-1]$ based on n, and registers the transaction $(Tran_i)$ in the blockchain.In the EC domain Parameter, $T = (G, n)$ plays an important role in the creation of the master key, so it is not open to the public, but only to the user. Therefore, only the $T = (p)$ required for verification is registered together.$d_i^{\wedge}$ is used in the ECC method to define it as a secret key, but the scenario proposed in this paper only creates a child key, thereby changing the role used in the original scenario. Therefore, the$Q_i^{\wedge}$, which plays an important role in creating the child Key, was kept secret and the $d_i^{\wedge}$ was opened to blockchain. The fact that it is difficult to guess the other key pair even if one key is released is characteristic of the public key algorithm.However, this method generates keys, including those$T = (G, n)$ between the public and secret keys, which provide greater safety than the public keys in the RSA method.In addition, a verification key for verifying $T = (G, n)$ are registered together to prevent attacks by reconfiguring EC parameters that satisfy $d_i^{\wedge}$ by disclosing $d_i^{\wedge}$.Figure 5 illustrates the use of ECC mechanism-based child keys for authentication scenarios.

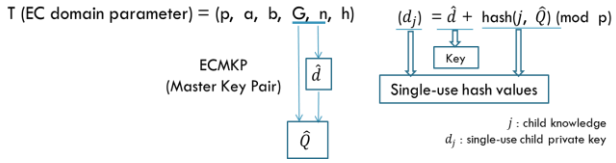

Fig 5. ECC mechanism used for design of the propsoed mechanism

Figure 5 illustrates the use of ECC mechanism-based keys for authentication scenarios.

$$d_i^{\wedge} \in [1, n-1] \quad (3)$$

$$EK = hash(G, n) \quad (4)$$

$$VK = hash(d_i^{\wedge}, EK) \quad (5)$$

$$Tran_i \leftarrow d_i^{\wedge}, P, EK \quad (6)$$

The service user uses $T = (G)$ and $d_i^{\wedge}$ selected during the service registration process to generate the master seed, $Q_i^{\wedge}$, as shown in Equation (7). Select any integer j and generate the j th child key. The user generates two hash values and requests authentication, as shown in Equation (8).

$$Q_i^{\wedge} = d_i^{\wedge} G \quad (7)$$

$$d_i^j \leftarrow \{ (d_i^{\wedge} + hash\ (j, Q_i^{\wedge}))mod\ p \} \quad (8)$$

$$HD_i^j \leftarrow \{ hash\ (d_i^j, EK) \} \quad (9)$$

$$HV_i^j \leftarrow \{ hash\ (j, Q_i^{\wedge}) \} \quad (10)$$

$$Request(HD_i^j,\ HV_i^j, EK) \quad (11)$$

C. Analysis of the proposed mechanism

Algorithm 1 shows the process of registering a key in a smart contract and then authenticating the user to a service provider.It shows the process of transmitting hash values only to verify that they are the same as the user who registered the keys in the blockchain.

Algorithm 1. Scenario for a Service Provider

1. *Get HV, HD*
2. *Get d, P*
3. *Calculate nc← $(d_i^{\wedge} + HV_i^j)modp$*
4. *Calculate NHD ← $hash(nc, EK)$*
5. *if NHD = HD, : compare(NHD, HD)*
6. *then permit action*
7. *else endif*

Algorithm 1 shows the authentication scenario on a channel shared with a service provider.Therefore, it is encrypted with the service provider's key, which may omit the verification phase of the EC domain Parameter. Algorithm 2 shows a scenario in which multiple service providers share authentication keys.

Algorithm 2. Scenario for Several Service Providers

1. *Get HV, HD, EK*
2. *Get d, P, VK*
3. *Calculate NVK ← $hash\ (d_i, EK)$*
4. *if NHD = VK,*
5. *then calculate nc← $(d_i^{\wedge} + HV_i^j)modp$*
6. *calculate NHD ← $hash(nc, EK)$*
7. *else if NDH = HD : compare(NHD, HD)*
8. *then permit action*
9. *else endif*

Algorithm 3 is a scenario that can be used for RBAC.A pair of roles is issued to the role issuer and one is registered and the remaining pair is digitally signed upon request for authentication.The contents included in the digital signature include time stamps and payments to prevent time difference attacks.

Algorithm 3. Scenario for Role-based Access Control

1. *Request role(p,q)_ ← role issuer issues*
2. *Register d, VK, role(p)*
3. *Request authentication : sign(HV, HD, EK,role(q) condition, time_stamp)q_{sc}*
4. *if VK is verified,*
5. *then create nc← $(d_i^{\wedge} + HV_i^j)modp$*
6. *create NHD from nc, and EK,*
7. *compare HD and NGD*
8. *submit condition*
9. *else endif*

D. Comparison with related studies

Table 1 compares the authentication scenarios of the studies involved with the authentication scenarios proposed in the paper.The related research was required to provide the same purpose as it intended to obtain certification from service providers by using blockchain without membership.The scenarios proposed in this paper were linked to the payment function without exposing any personal information.It also designed autonomous management functions to allow users to manage their own keys.The proposed scenario implemented a lightweight scenario with simplified parameters.In the related study, the scenario in which five entities were involved was designed to involve only three entities in the scenario proposed in the paper.The relevant research is based on a trust-based payment system despite the use of blockchain, while the scenarios proposed in this paper have taken advantage of the "trustless" prepayment based on smart contract.

Table- 1: Comparison with blockchain-based service

Blockchain-based service	Multi-purpose Authentication Mechanism
User to Server *Virtual ID registration* - *User's virtual ID, User's public key* Server to Blockchain *Register ID and Key to Blockchain* - *User's virtual ID, User's public key* - *Signature* User to Partner *Service access request* - *User's virtual ID, nonce* - *Signature* Partner to Blockchain *Blockchain lookup* - *User's virtual ID* - *User's public key* Partner to User *Request authentication* - *User's virtual ID, nonce+1* - *Encrypt partner's public key* User to Partner *Response authentication* - *User's virtual ID, nonce+2* - *Encrypt virtual ID and nonce+2*	User to Blockchain *Register key* $Tran_i \leftarrow d_i^{\wedge}, P, EK$ User to Blockchain *Request three hash values* $d_i^j \leftarrow \{ (d_i^{\wedge} + hash\ (j, Q_i^{\wedge}))mod\ p\}$ $HD_i^j \leftarrow \{ hash\ (d_i^j, \mathrm{EK})\}$ $HV_i^j \leftarrow \{ hash\ (\mathrm{j}, Q_i^{\wedge})\}$ $Request(HD_i^j, HV_i^j, EK)$ Partner to Blockchain *Get key* $NVK \leftarrow hash\ (d_i, \mathrm{EK})$ $EK = hash(G, n)$ $Compare(NVK, EK)$
Partner requests extra information for the user	*Don't distribute user information*
They used blockchain only as a bridge	*Blockchain acts as a platform for a trustless contract.*

Table 2 compares the scenarios proposed in this paper with the smart contact-based RBAC introduced in the second related study.In the related research, the role publisher registered the role directly in the smart application for transparency of the role.This is the author's intention to provide the transparency of the role because of the blockchain's characteristics that become the registrar's digital signature upon registration.To achieve the same purpose, the role issuer issues the role, signs it with its signature key, and forwards it to the user, before the user registers his or her own authentication key and role together.Because the user's digital signature will be registered at the time of registration, the service provider will be able to verify the user in future service authentication. It can be verified by the role issuer's public key, so that both role transparency and user impersonation can be identified at one time, and even the payment function can be linked.Therefore, the objective can be simply achieved without using the Challenge-Response protocol, which was proposed by the related study.It also allows users to manage their own security information. It can take advantage of both the ability to make payments and the integrity and unmodifiable evidence that is the most important function of smart apps. Table 2 illustrates this comparison scenario

Table- 2: Comparison with Role-based Access Control for Smart Contract

Role-based Access Control for Smart Contract	Multi-purpose Authentication Mechanism
User to Role Issuer *Request role creation* Role Issuer to Smart Contract : *Role Creation* *Create contract* *Execute SC* *Publish SC details* User to Service Provider : *Authentication* *Declare EOA* *Check Information* *Challenge process* *Verify Response* Service Provider to User : *Challenge Response protocol* *Send Message(m)* *Signature*	Role Issuer to User : *Role Creation* *Send digitally signed role* - *Role signed by key* User to Blockchain *Register key* - *Key* User to Service Provider *Request service provision* - *Role, key, condition* - *sign(HV,HD,EK,role(q) condition, time_stamp)q_{sc}* Partner to Blockchain *Verify key and role* Key, role, condition
They used blockchain only as a role storage.	*Provide authentication and role-based control at a time without unnecessary processes*
They implemented additional, unnecessary authentication protocols.	*Blockchain acts as a platform for a trustless contract.*

E. Contribution

Privacy and Anonymity

The user did not disclose any of his or her information and constructed the algorithm using a mechanism to prove that he was the owner.Seed is reserved only by himself, so only himself can succeed in certification. In addition, since personal information is not registered, it can be used in a variety of areas as it can be reassuring to expose information and other attacks.It was allowed to verify and realize a specific role without disclosing the mapping with customer information.To realize the privacy of the customer's information, we sought complete.

Multi-Purpose Certification

By using the framework of mechanisms that provide anonymity and privacy for various purposes, authentication to realize multiple objectives can be provided.Users can be authenticated to associate accounts with payments and can be utilized in various RBAC models by using them for role-based purposes.One

service provider with a contact can encrypt and use each other without key distribution, allowing more important information to be shared.In addition, multiple service providers can be used for various purposes, as well as to defend against attacks that may occur by revealing keys.

IV. CONCLUSION

In this paper, we proposed a blockchain-based mechanism that performs authentication functions for various purposes. One key registration can be certified by multiple service providers, and trustless has been realized by making full use of the strengths of blockchain, such as integrity and transparency.It is also designed to be applied to RBAC models using digital signatures, etc.Designed to allow users who are masters of the master seed to manage their own security.The authentication key was registered using the smart contact, and anonymity was provided using only the disposable hash value as the authentication parameter.A more secure method was proposed together with a specific service provider if the smart contact channel was used.It also analyzed various attacks and proposed countermeasures against them. Research is also needed on mutual authentication linking the payment function of smart applications with existing services while providing anonymity in the future

ACKNOWLEDGMENT

This work was supported by the Next-Generation Information Computing Development Program through the National Research Foundation of Korea, Republic of Korea (Grant no.: NRF-2017M3C4A7069432).

REFERENCES

1. NOVO, Oscar. Blockchain meets IoT: An architecture for scalable access management in IoT. IEEE Internet of Things Journal, 2018, 5.2: 1184-1195.
2. OUADDAH, Aafaf; ABOU ELKALAM, Anas; AIT OUAHMAN, Abdellah. FairAccess: a new Blockchain-based access control framework for the Internet of Things. Security and Communication Networks, 2016, 9.18: 5943-5964.
3. LEE, Jong-Hyouk. BIDaaS: Blockchain based ID as a service. IEEE Access, 2018, 6: 2274-2278.
4. CHRISTIDIS, Konstantinos; DEVETSIKIOTIS, Michael. Blockchains and smart contracts for the internet of things. Ieee Access, 2016, 4: 2292-2303.
5. HAMMI, Mohamed Tahar, et al. Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. Computers & Security, 2018, 78: 126-142.
6. KISHIGAMI, Junichi, et al. The blockchain-based digital content distribution system. In: 2015 IEEE Fifth International Conference on Big Data and Cloud Computing. IEEE, 2015. p. 187-190.
7. CRUZ, Jason Paul; KAJI, Yuichi; YANAI, Naoto. RBAC-SC: Role-based access control using smart contract. IEEE Access, 2018, 6: 12240-12251.
8. OUADDAH, Aafaf; ELKALAM, AnasAbou; OUAHMAN, AbdellahAit. Towards a novel privacy-preserving access control model based on blockchain technology in IoT. In: Europe and MENA Cooperation Advances in Information and Communication Technologies. Springer, Cham, 2017. p. 523-533.
9. HANKERSON, Darrel; HERNANDEZ, Julio López; MENEZES, Alfred. Software implementation of elliptic curve cryptography over binary fields. In: International Workshop on Cryptographic Hardware and Embedded Systems. Springer, Berlin, Heidelberg, 2000. p. 1-24.
10. GURA, Nils, et al. Comparing elliptic curve cryptography and RSA on 8-bit CPUs. In: International workshop on cryptographic hardware and embedded systems. Springer, Berlin, Heidelberg, 2004. p. 119-132.
11. BOS, Joppe W., et al. Elliptic curve cryptography in practice. In: International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2014. p. 157-175.
12. GUTOSKI, Gus; STEBILA, Douglas. Hierarchical deterministic Bitcoin wallets that tolerate key leakage. In: International Conference on FinancialCryptography and Data Security. Springer, Berlin, Heidelberg, 2015. p. 497-50

AUTHORS PROFILE

Yong Joo Lee received the M.S., and Ph.D. from Chungbuk National University, Korea. She had been a senior researcher at ETRI in Korea from 2001 to 2009. She is currently a chief researcher at IT global research institute of KNUT in Korea. Her research interest includes security, cryptology, blockchain, and machine learning.

Keon Myung Lee received B.S., M.S., and Ph.D. from KAIST, Korea. He is a professor at Department of Computer Science, Chungbuk National University. His research interest includes machine learning, data mining, big data, blockchain, and artificial intelligence applications.