Blockchain-Based User Authentication with Anonymity for Internet of Things Applications

[1]YongJoo Lee, Keon Myung Lee, Sang Ho Lee

Abstract: ***Security threats that target identities (IDs) have increased considerably in recent years. Various network attacks attempt to discover IDs that can be used in future attacks to obtain private information. In this paper, we propose a blockchain-based user authentication approach that can be used by various end-users for internet of things (IoT) applications. The proposed approach uses single-use authentication parameters and does not require any private information. It is based on the child-key mechanism of the hierarchical deterministic (HD) wallet. The HD wallet is accepted as a standard of Bitcoin and is in turn based on the elliptic curve digital signature algorithm. The authentication parameters of the proposed approach were created using the HD wallet mechanism. The transaction of the authentication key stored in the distributed ledgers of the blockchain could be shared by various IoT servers for subscription to different services, and the user account of the blockchain could be connected to the IoT servers for payments without a membership procedure. We used only hash values for an authentication request to protect against network attacks. The proposed approach could decrease the system load by using a lightweight feature with few parameters and simplify the approach without the need for additional procedures by IoT servers. We verified that the security requirements of the proposed approach were satisfied, by analyzing the transmitted parameters. Furthermore, we evaluated the security vulnerabilities from various threats and analyzed attack scenarios. Thus, we propose that the authentication servers verify an original EC domain parameter hash value and the hash reuse to protect against fake attacks by network sniffing and spoofing. We have also summarized the originality and the characteristics of the proposed research by comparing it with closely related studies and concluded with a future research guide.***

Keywords: Blockchain, Privacy, Authentication, Peer to Peer, Identity, Elliptic curve cryptosystem

I. INTRODUCTION

The IoT is a growing paradigm with technical, social, and economic significance that comprises a wide ecosystem of interconnected services and devices such as sensors, consumer products and cars. Industry 4.0 and industrial internet of things (IoT) are frequently and rightfully associated with IoT focusing on digitizing industries. Industry 4.0 focuses on creating smart products, procedures, and processes, and the industrial IoT well manages the complexity and can be used to more efficiently manufacture goods. Human beings, machine devices, and resources communicate as naturally as in a social network [1]. Devices with sensors collect sensitive environmental information and provide it to authorized machines for monitoring or analysis. The analyzed information provides the basis for decisions to take various actions, including actuating sensors, which can have important consequences. Such systems are vulnerable to cyber-attacks, since the deployment of many sensors in an unprotected space introduces the attack space, and the sensors are beyond the traditional perimeter defense control mechanisms. Manufacturing data sourced from different devices are notably attractive to cybercriminals [2, 3]. For example, cybercriminals may financially benefit from theft by selling data to third-parties, and malicious users, who intend to create chaos, can take over various sensors and report false data to the system. It is critical that such systems provide at least data and device integrity, authentication, and data access control [4, 5]. Blockchain is essentially a distributed database or digital ledger of transactions that have been shared among participating parties. Two important characteristics of the blockchain are distributed consensus and anonymity. The privacy protection measure is not robust in blockchain, although many blockchain systems take a measure [6]. We aim at providing a user authentication mechanism with anonymity and privacy with a light-weight feature for our research. The authentication parameters do not include private details and the transactions stored in a distributed ledger of blockchain can be shared by various IoT servers for service subscriptions. Here we will propose a new authentication approach, describe a use-case scenario, and analyze the proposed approach. The remainder of this paper is organized as follows. Section2 proposes the proposed user authentication procedures. Section 3 analyzes the procedure security and attacks. Section 4 concludes the study and provides future direction.

Revised Manuscript Received on January 03, 2019.

YongJoo Lee, Department of Computer Science, Chungbuk National University, Chungdae-ro 1, Heungdeok-ku, Cheongju, Chungbuk 361-763, Korea,

Keon Myung Lee, Corresponding author, Department of Computer Science, Chungbuk National University, Chungdae-ro 1, Heungdeok-ku, Cheongju, Chungbuk 361-763, Korea,

Sang Ho Lee, Department of Computer Science, Chungbuk National University, Chungdae-ro 1, Heungdeok-ku, Cheongju, Chungbuk 361-763, Korea,

II. ANONYMITY-PRESERVING USER AUTHENTICATION

This section describes the requirements of the proposed approach, a network configuration and a detailed description. The proposed approach consists of a registration, request of authentication, and authentication procedure.

2.1. The proposed User Authentication

In the blockchain-based decentralized network, many IoT servers can share distributed ledgers in a blockchain to keep

The integrity of authentication data without a trusted third party. When a user needs to subscribe a service from a IoT server, the user can ask to be authenticated by using the parameters stored in the ledgers. Storing any private data in each server can be a big threat in the centralized network. Here, we aim at providing a light weight user authentication approach which provides anonymity and privacy. Anonymity in a user account of a blockchain will be connected to the proposed approach. The approach is based on a decentralized peer-to-peer (P2P) protocol where the parameters are accessed by various IoT servers without a trusted third party. Users can be connected to other IoT servers by the blockchain account for a payment. When the user wants to change the registered parameters, the user does not need to notify any IoT server of the change because a blockchain manages transactions in a chronological order by an internal time server. Figure 1 shows the simplified structure of the proposed approach.

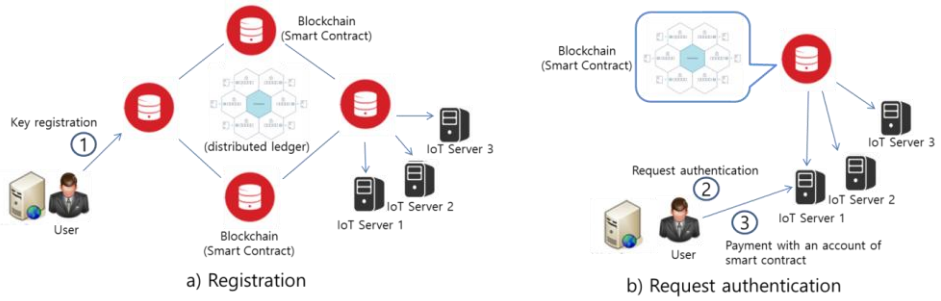


Figure 1. Simplified procedures

2.1.1. Detailed Description

The degree of privacy is expressed through indistinguishable transactions on the blockchain. Although many blockchain systems take measures to protect privacy, it is not robust [7, 8]. Fixed IDs or keys can leave a clue for user private details in a specific service. A specification of hierarchical deterministic (HD) wallets was proposed in 2012 and accepted as Bitcoin standard BIP32, which generates wallet keys from a single master seed [9, 10]. The HD wallets based on Elliptic Curve Digital Signature Algorithm (ECDSA) which is based on the algebraic structure of elliptic curves over finite fields [11, 12]. Bitcoin uses elliptic curve signatures on curve secp256k1. We use the HD wallets to create a child key from a master seed and produce hash values from the child key. The proposed approach consists of three procedures: registration, request of authentication, and authentication. ECDSA uses Elliptic Curve (EC) domain parameters to generate a master key pair that strengthens ECDSA based approach. Figure 2 shows a brief procedure.

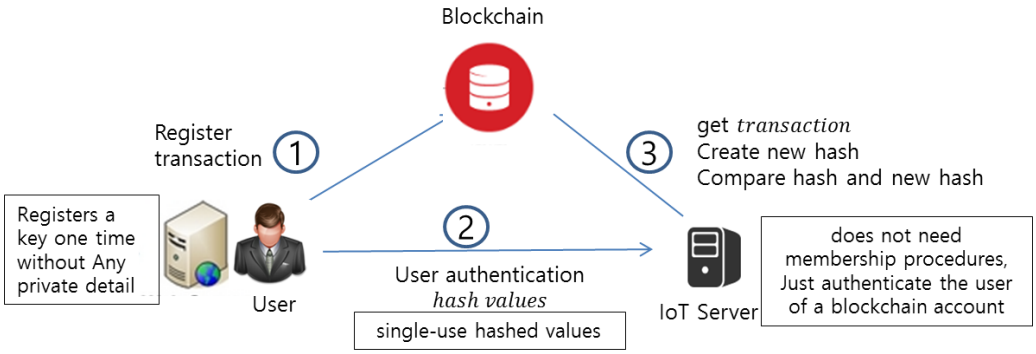


Figure 2. Proposed procedures

- ***Registration***

Let I be the set of users; J is the set of child key; F_p is the prime finite field, which contains p elements, where each element is a set of integers $\{0, 1, 2, \ldots, p-1\}$; EC is the domain parameter; T over F_p is a sextuple, $T = (p, a, b, G, n, h)$. $T(G)$ is an elliptic curve base point, which is a generator of the elliptic curve with large prime order $T(n)$, and n is the integer order of $T(G)$. We use · to denote the elliptic curve point multiplication by a scalar. We assume that all parameters are appropriately generated and verified. A user selects a cryptographically secure random integer, $d_i^{\wedge}$ in the interval $[1, n-1]$. The corresponding public key is $Q^{\wedge} = (d^{\wedge} \cdot G)$. The elliptic curve master key pair ($ECMKP_i$) is

$$ECMKP_i = (d_i^{\wedge})(Q_i^{\wedge}) \quad (1)$$

The user generates a verifying key, VK,

$$EK = hash(G, n) \quad (2)$$

$$VK = hash(d_i^{\wedge}, EK) \quad (3)$$

The user registers transaction, $Tran_i$,

$$Tran_i \leftarrow (d_i^{\wedge}, VK) \quad (4)$$

We do not use an encryption algorithm for $Tran_i$, because $Tran_i$ consists of a random integer and a hash value. The registration is a blockchain transaction that is broadcast to every node. A new block is added only after a successful verification. The transaction address, which is the hash value of the public key, is returned as a transaction ID. $Tran_i$ will be shared by various IoT servers which want to authenticate the user.

- ***Request of authentication by a user***

When the user wants to get authenticated by an IoT server for a service subscription, hash values from a master key pair are generated. The user generates a pseudorandom sequence of a child key under a given ECMKP using only knowledge j and calculates two hash values (HV and HD),

$$HV_i^j = hash(j, Q_i^{\wedge}) \quad (5)$$

$$d_i^j = (d_i^{\wedge} + HV_i^j)(mod p) \quad (6)$$

$$HD_i^j = hash(d_i^j, EK) \quad (7)$$

The user transmits three hash values, (HV, HD, and EK) to the IoT server. The first hash value, HV will be transmitted to enable the server to create a child key under the same master seed. The second, HD is for comparison with the received value and the calculated value. The last, EK will be used to verify the original EC domain parameters. The master key, $Q^{\wedge}$ is an important source to create a child key. We transmit the hashed value of j and $Q^{\wedge}$ to protect against network attacks. The result of the hash function from a bit change is totally different; hence, j cannot be reused even after a single usage.

- ***Authentication by IoT Server***

When the user asks an authentication to a IoT server with the hash values, the sever gets $Tran_i$ of the user from the ledger of a blockchain and generates NVK and nc from HV and $d_i^{\wedge}$. Then the server calculates NHD from nc,

$$NVK \leftarrow hash(d_i^{\wedge}, EK) \quad (8)$$

$$nc = (d_i^{\wedge} + HV_i^j)(mod\ p) \quad (9)$$

$$NHD \leftarrow hash(nc, EK) \quad (10)$$

The server compares NVK and VK to verify the original parameter set. If they are the same, the server can confirm that the received hash values were generated from the same EC domain parameters. Next, the server compares NHD with HD to authenticate the user. If they are the same, the server can confirm that the new child key was generated from the same master seed which the user created for the first time. Then the IoT server can authenticate the user, who requested the authentication, as the owner of the account. If authenticated, the user registers the hash value HV_i^j to the blockchain to prevent a hash reuse attack. The hash value is publicly known as a one-way function which is cryptographically secure. The result of the hash function from a bit change is totally different; hence, j cannot be reused even after a single usage. If blockchain is private type and located in a secure place, registering the hash value is not essential. Figure 3 (a) shows a sequence diagram among a user, an IoT server and a blockchain and figure 3 (b) shows a flow chart in the IoT server which fulfills an authentication mechanism by receiving parameters from the user and retrieving parameters from the blockchain [17,18].

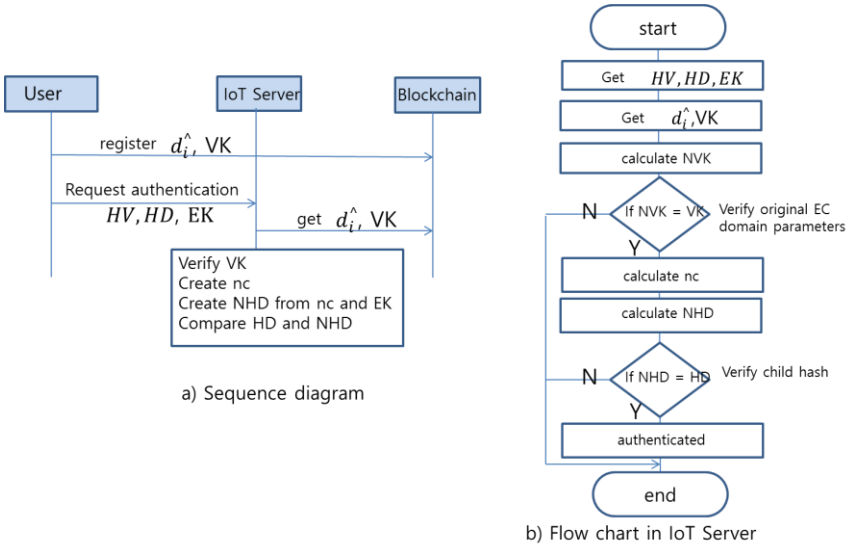


Figure 3. Sequence diagram and flow chart

2.2. Use-case Scenario for Smart Factory

We describe a use-case scenario to show how the proposed approach can be adopted in smart factory as an IoT application. We define user groups and a network configuration for the end-users in smart factory.

2.2.1. Definition of role-based user group

We define user groups according to their roles to develop a use-case for industrial IoT applications as shown in table 1. Authentication methods are assigned to each user group. Identity/password (ID), authentication token (Token), media access control (MAC) address-based scheme, certificate-based scheme (Cbs), and cryptography-based scheme (Cry) are described as an authentication method. One-way and Mutual authentications are described as an authentication direction as shown in Table 1. We propose a cryptography-based approach for one-way user authentication which is commonly used for high-security-level applications in IoT applications.

Table 1: Role-based user group definitions in Smart Factory

User-group	Role	Type	Method	Priority
Managers	Monitoring	One way	ID, Cbs	Security, efficiency
	System Command	One way	ID,Cbs,Cry	Security
	Security Control	One way	ID, Cbs, Cry	Security
Applications	Event, command	One way	Token, MAC	Efficiency
End_users	Customer, designer, provider, manufacturer	One way	ID, Cbs, Cry	Security, privacy, anonymity
		Mutual		

2.2.2. Decentralized network configuration for Smart Factory

Using the defined role-based user groups from Table 1, we consider machine-to-machine (M2M) devices, network, and application domains with end-users in the blockchain-based decentralized environment. The Internet of service in the application domain is connected with an open API, which is provided by the IoT Platform. The end users in the application domain can be connected to the cloud or IoT platform. The precise architecture depends on the development. IBM uses a cloud infrastructure for blockchain services to track high-value items as they move across supply chains. The service capabilities of the IBM Watson IoT platform enable users to add IoT data to private distributed ledgers that can be included in shared transactions [6]. We assume an interoperable private blockchain. Devices with sensors and actuators are connected with M2M communication. The IoT platform, cloud, and blockchain are located in the network domain. Figure 4 shows the network configuration with existing implementable technologies.

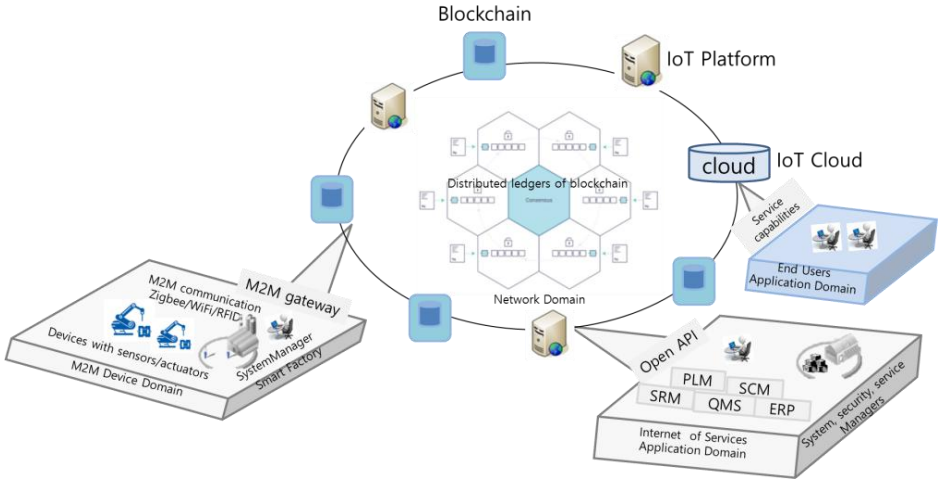


Figure 4. Decentralized Network Configuration

III. RESULTS AND DISCUSSION

We evaluated the security requirements and vulnerabilities. We also analyzed possible attack scenarios including network and fake attacks. We compared the proposed approach with closely related studies to show originality and differences in this section.

3.1. Security Evaluation

- **Vulnerability from the parameter exposure**: We analyzed vulnerabilities from registering $d^{\wedge}$ to the blockchain. $d^{\wedge}$ is chosen by n and, $d^{\wedge}$ lies between 1 and $n-1$. Attackers need $Q^{\wedge}$ to create the authentication parameters. The attackers need elliptic curve base point $T(G)$ to create $Q^{\wedge}$. If an attacker can easily guess $T(G)$, it can be risky. But the probability to guess the EC domain parameter $T(G)$ without any knowledge is $P = \frac{1}{2^x}$, where x is the bit size of the EC domain parameters $T(G)$. The common bit size of n is from 163 to 512, and $T(G)$ is bigger than n. The value is extremely small. If we view it from a different attack point, the attackers can generate a new EC domain parameter set which satisfy the original $d^{\wedge}$, because $d^{\wedge}$ is chosen between 1 and $n-1$. Hence, we register VK to verify the original EC domain parameters from the hash value of T (n, G),

$$hash(d_i^{\wedge}, hash(G, n)) \neq hash(d_i^{\wedge}, hash(G', n'))$$

- **Vulnerability of the authentication parameters**: the user transmitted three hashed values by a P2P network in the approach. A hash value is one-way and the original of the hash value cannot be extracted. Although, attackers successfully capture the authentication parameters, they cannot calculate the original keys,

$$d_i^{\wedge}, EK \times\leftarrow VK \leftarrow hash(d_i^{\wedge}, EK)$$
$$(j, Q_i^{\wedge}) \times\leftarrow HV_i^j \leftarrow hash(j, Q_i^{\wedge})$$
$$d_i^j, EK \times\leftarrow HD_i^j \leftarrow hash(d_i^j, EK)$$

Currently, secure hash algorithm (SHA)-256 is accepted by many blokchains including Bitcoin. Collision was detected in SHA-1, but SHA-256 included in SHA-2 is publicly known as secure.

3.2. Analysis of Attack Scenarios

We analyzed possible attacks during the user authentication performance. We supposed attacks of network sniffing, fake user, and malicious user.

Case 1: Network Attack: Data sniffing/spoofing

The sniffing attack is the interception of data by capturing the data traffic using a sniffer, which is an application to capture network packets. We suppose that the attackers attemptto obtain thehashed values. Figure 5(a) shows the data sniffing attack scenario.

Figure 5. Network attack scenario

The parameters transmitted to the IoT server are only hashed values. Although, attackers successfully obtain the hash values by network attacks, they cannot extract the original parameters.

Case2: Fake attack by the same hash values

An attacker can pretend to be a fake user by sending the same data captured fromnetwork attacks instead of trying to extract the original parameters. Suppose that the attackers get thehash values from the network attacks and attempt to be a fake user by sending the same data in the next transaction. Figure 5(b) shows the fakeattack scenario.We do not reuse a child knowledge, j, and the hash value from the same j is thrown out in the proposed approach. A user saves the hash value HV_i^j generated from a random integer j after a single use. The other party checks the reuse of the hash value before accepting the authentication. Hence, even if a fake user sendsthe same data, the authentication cannot be accepted.

Case3: Fake attack by a new parameter set

Attackers can try to get authenticated by the original $d^{\wedge}$ and a new parameter set. Suppose that the attackers generate new EK' from G' and $d^{\wedge}$, new HV' from $Q_i^{\hat{}\prime}$, and new HD' from EK' and d'. This attack can be easily protected by comparing VK' with VK which is generated by the original n and G of the EC domain parameter because the original n and G are included in the hash value, EK. The EC domain parameter, n and G cannot be extracted by the attackers. Figure 6 shows the attack procedure by a new parameter set.

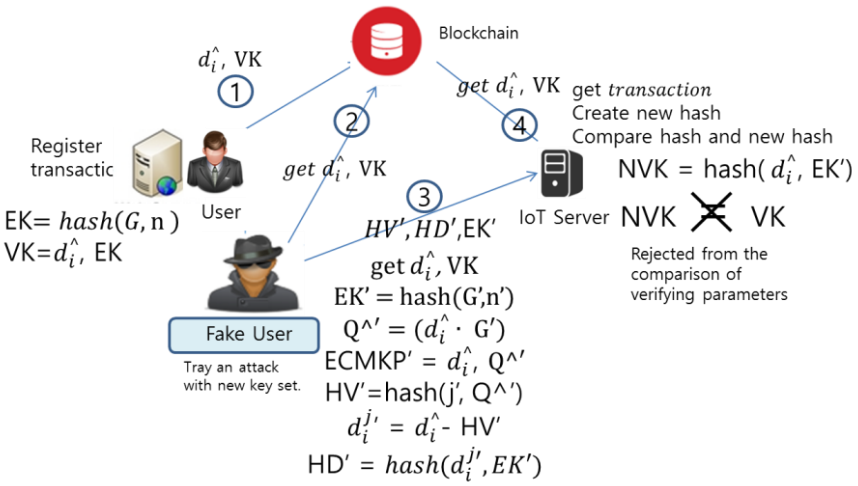
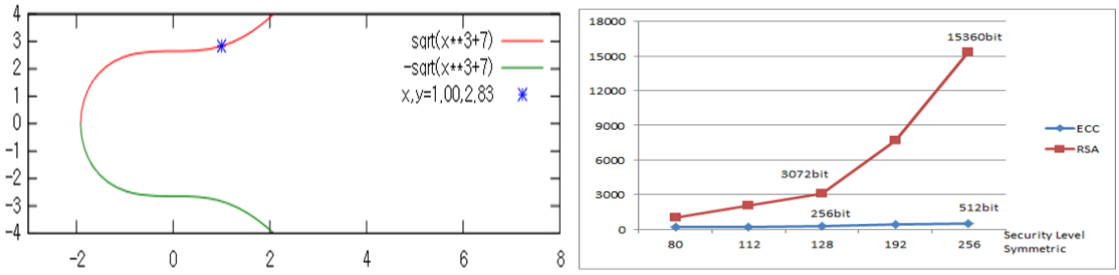


Figure 6. Analysis of the fake attack scenario by a new parameter set

3.3. Comparable ECC and RSA Security Level

The public key of the ECC-based scheme represents a point on a particular Elliptic Curve (EC). It contains an identification byte, a 32-byte X coordinate, and a 32-byte Y coordinate defined in secp256k1. Figure 7 (a) shows a point on a simplified EC used by Bitcoin, $y^2 = x^3 + 7$, over a field of contiguous numbers. The public key size can be reduced by 50% without changing any fundamentals by dropping the Y coordinate. Only two points along the curve share any particular X coordinate, therefore a 32-byte Y coordinate can be replaced with a single bit to indicate whether the point is on the top side or the bottom side [8]. Figure 7 (b) shows comparable key sizes for ideal symmetric ECC-based and RSA-based schemes. Smaller key sizes are required to provide the equivalent security level [6, 7].



a) EC of SECP256k1 b) ECC and RSA Key sizes

Figure 7. Comparable key sizes

3.4. Comparison and discussion

- Evaluation of the approach requirements: The proposed approach is designed for being used by multiple servers and connected with an account for a payment. The approach has a light-weight feature with few hash parameters. Table 2 shows the evaluation for each requirement.

Table 2: Evaluation of requirements

Requirements	Evaluation
Anonymity	A user does not need to reveal an identity or private information. The authentication parameters are hash values and do not include user information. Therefore, anonymity in a blockchain account is connected to IoT servers.
Privacy	
Light-weight	We use three parameters for an authentication request. An authentication procedure is one-step without an additional membership procedure by IoT servers for service subscription.
Multiple-server	Transactions stored in distributed ledgers can be shared by various IoT servers for the same user account.
Against ID attacks	The proposed approach does not reveal any identity to protect against ID attacks.
Same account for payment	The user account of a blockchain is connected to IoT servers for a payment of service subscriptions without an additional procedure.
Control by users	A user can change the authentication parameters stored in a distributed ledger of the blockchain and do not need any announcement or notification.

- Comparison of related studies: We listed the closely related studies hereto verify the necessity and originality of the proposed research in table 3. In the proposed approach, the user can subscribe many services by various IoT servers without subscription with private data, through one-time parameter registration to blockchain. The users do not depend on the security of a specific system or a trusted third party because the users choose and register the authentication data by themselves. Authentication parameters do not include any private information to provide anonymity.

Table 3: Comparison of related studies

	Difference	Limitation
Aref et al. [13]	Application domain security provide authentication between entities: sensitive data is recommended to be stored in the secure storage of the M2M system, security depends on the trusted and centralized M2M system.	Not based on decentralized control. No user control in security and storage. Need a trusted third party.
Sergio et al. [14]	Capability-based security approach to manage access control in the IoT. A capability is a communicable token of authority. It was developed for a centralized IoT environment.	Need a trusted third party and centralized storage. No anonymity and privacy mechanisms.
Jason et al. [15]	Role-based access control uses smart contract for challenge and response protocol. The user role is issued by the trusted organization, and other users verify the role through the role-issuing organization for every service as a trusted third party.	No privacy and anonymity mechanisms. System loads to verify the roles every time. Not light-weight
Yanqi et al. [16]	Blockchain-based fair payment. It uses the broker-less publication and subscribes a protocol by the blockchain to exchange private data.	No mechanisms for privacy and against ID attacks.

IV. CONCLUSION

We proposed a user authentication approach that can be useful in the convergence of blockchain and IoT applications. We focused on providing anonymity and privacy to protect the private information of users. We created single-use parameters by using the HD wallet child-key mechanism based on ECDSA. The authentication parameters registered in the distributed

ledgers of the blockchain could be shared by various IoT servers for service provisions without additional membership procedures. The user account of the blockchain was connected to the IoT servers for the payments of service subscriptions. A user who registered the authentication parameters to the blockchain did not have to notify the IoT servers after changing the parameters, because the transactions in the blockchain were updated by timestamps. The proposed approach decreased the system load by using a lightweight feature with few parameters for an authentication request. We accepted the blockchain technology to enable entities to act in a decentralized environment without a trusted third party and connect a user account to an IoT service for payments. Then, we evaluated the procedure security and analyzed the attacks. We confirmed that the proposed approach was secure from the analyzed attacks. Although the proposed protocol offers several advantages, it has room for improvement. In the future, we intend to develop a detailed access model with a policy structure. A mutual authentication approach between two entities for cooperation should also be developed to the required security level with decentralized control.

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. Florian T, Bjorn S. Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. IEEE Communications Surveys & Tutorials. 2016 third quarter;18(3):2084-2123.
2. Oscar N. Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT. IEEE Internet of Things Journal, 2018 Apr;5(2):1184-1195.
3. Nikolay T, Igor R. Blockchain-based platform architecture for industrial IoT. Open Innovations Association (FRUCT), 2017 21st Conference of IEEE, 2017:321-329.
4. Gaby GD, Jordan M, Matea M, Praneeth BM. Ancile:Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology. Sustainable Cities and Society. 2018 May;39:283-297.
5. Chao L, Debiao H, Xinyi H, Kim K, Raymond C. BSein:Ablockchain-based secure mutual authentication with fine-grained access control system for industry 4.0. Journal of Network and Computer Applications. 2018 May;116(15):42-52.
6. Nir K. Can Blockchain Strengthen the Internet of Things?. IEEE IT Professional. 2017;19(4):68-72.
7. Xiaoqi L, Peng J, Ting C, Xiapu L, Qiaoyan W. A survey on the security of blockchain systems. Future Generation Computer Systems, 2017.
8. Peng J, Fuchun G, Kaitai L, Jianchang L, Qiaoyan W. Searchain: Blockchain-based private keyword search in decentralized storage. Future Generation Computer Systems. 2017.
9. Bitcoin. HD Wallet. Available from:https://bitcoin.org/en/developer-guide#signing-only-wallets
10. Gus G, Douglas S. Hierarchical deterministic Bitcoin wallets that tolerate key leakage. International Conference on Financial Cryptography and Data security. 2015Jan;LNCS(8975):497-504.
11. Daniel RLB. Sec 1:Elliptic Curve Cryptography. Standard for Efficient Cryptography. 2009May. Available from:http://www.secg.org/sec1-v2.pdf
12. DanielRLB. Sec 2:Recommended Elliptic Curve Domain Parameters. Standard for Efficient Cryptography. 2010 Jar. Available from:http://www.secg.org/sec2-v2.pdf
13. Aref M, Internet of things standards: who stands out from the crowd?.IEEE Communications Magazine. 2016 Jul;54(7):40-47.
14. Sergio G, Salvatore P, Domenico R. A capability-based security approach to manage access control in the internet of Things. Mathematical and Computer Modeling. 2013 Sep;58(5):1189-1205.
15. Jason PC, Yuichi K, Naoto Y. RBAC-SC:Role-based Access Control Using Smart Contract. IEEEACESS. 2018 Mar;6:12240-12251.
16. Yanqi Z, Yannan L, QilinM, BoY, YongY. Secure Pub-Sub: Blockcain-based Fair Payment with Reputation for reliable cyber physical systems. IEEEACCESS. 2018 Mar;6:12295-12303.
17. Nallapaneni Manoj Kumar, **Pradeep Kumar Mallick**, "*Blockchain technology for security issues and challenges in IoT*", Elsevier Procedia Computer Science Journal, Volume 132, Pages 1815-1823 , 2018, ISSN:1877-0509, UGC Sl No: 46138 and 48229, DOI: https://doi.org/10.1016/j.procs.2018.05.140.
18. Nallapaneni Manoj Kumar, **Pradeep Kumar Mallick,**" *The Internet of Things: Insights into the building blocks, component interactions, and architecture layers*", Elsevier Procedia Computer Science Journal , Volume 132, Pages 109-117, 2018, ISSN:1877-0509, UGC Sl No: 46138 and 48229 DOI: https://doi.org/10.1016/j.procs.2018.05.170.