IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 June 2022, accepted 12 July 2022, date of publication 15 July 2022, date of current version 21 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3191414

RESEARCH ARTICLE

On the Design of a Privacy-Preserving Communication Scheme for Cloud-Based Digital Twin Environments Using Blockchain

SEUNGHWAN SON[1], DEOKKYU KWON[1], JOONYOUNG LEE[1], (Student Member, IEEE), SUNGJIN YU[1,2], (Student Member, IEEE), NAM-SU JHO[2], AND YOUNGHO PARK[1,3], (Member, IEEE)

[1]School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, South Korea
[2]Electronics and Telecommunications Research Institute, Daejeon 34129, South Korea
[3]School of Electronics Engineering, Kyungpook National University, Daegu 41566, South Korea

Corresponding author: Youngho Park (parkyh@knu.ac.kr)

This work was supported by the Korean Government through the Electronics and Telecommunications Research Institute—ETRI (Core Technology Research on Trust Data Connectome) under Grant 20ZR1300.

ABSTRACT Digital twin technology is recently in the spotlight because of its potential applicability in business and industry. In digital twin environments, data generated from physical assets are transmitted to a remote server, which performs simulations through digital twins in a virtual space. Then, the simulation results can be shared with the data owner or other users. However, several challenges exist in the application of digital twin technology in the real world. One of the main challenges involves determining how to share real-time data for the simulation and how to share the simulation data securely. The data generated from physical assets may include sensitive information from data owners, and the leakage of data to an adversary can cause serious privacy problems. Moreover, the sharing of data with other data users should also be considered to maximize the availability of digital twin data. To resolve these issues, we propose a system model for the secure sharing of digital twin data. The proposed system model uses cloud computing for efficient data sharing and blockchain for data verifiability. We also propose communication schemes for the proposed model to guarantee privacy preservation and data security in wireless channels. We analyze the security of the proposed protocol using informal methods and formal methods such as BAN logic and the AVISPA simulation tool. Furthermore, we compare the proposed protocol with related protocols and demonstrate that the proposed scheme is applicable to digital twin environments.

INDEX TERMS Digital twin, mutual authentication and key agreement, blockchain, cloud computing, BAN logic, AVISPA.

I. INTRODUCTION

Digital twin technology involves performing simulations using a clone of a physical object that is represented in virtual space. The concept of simulation using a clone in a virtual space was first presented by Grieves and Vickers in 2002 [1], and NASA called the concept a digital twin in 2010 [2]. However, it has not been specifically discussed because of the lack of technology to realize digital twins. In particular, it is difficult to transmit and receive big data generated by digital twins in real time. With the rapid advances in communication technology, digital twin services are expected to become available in the near future. Digital twin technology has generally been studied in manufacturing, construction, and space industries. Recently, the application range of digital twin technology has expanded to mobile devices and the

The associate editor coordinating the review of this manuscript and approving it for publication was Zheng Yan.

Internet of Things (IoT); for example, when applied to vehicular environments, autonomous driving can be realized, and when applied to medical environments, detailed and precise remote treatment can be performed. In addition, the potential services are expected to be limitless if this technology can be applied to real environments.

The most common method of implementing digital twin services is to use cloud computing. In cloud-based digital twin environments, data generated from physical assets are transmitted to a cloud server that performs simulations using the data through digital twins in a virtual space [3], [4]. When the simulation results are generated, the data owner can receive it, or other users can use it upon request. In this process, most of the data are transmitted through wireless channels, and the data may include content related to owners' privacy. Therefore, it is necessary to develop a secure method for sharing transmitted data.

One of the main considerations for secure digital twin data sharing is the verification of data integrity [5], [6]. A data user can request data from a cloud server as needed, but the user cannot verify the integrity of the data in typical cloud computing environments. Data users may not fully trust the data, which can negatively affect network reliability. Furthermore, digital twin services are expected to be realized in the sixth generation (6G) wireless environment, and wireless channels are considered vulnerable to various attacks such as man-in-the-middle (MITM), replay, and ephemeral-secret-leakage (ESL) attacks. In addition, security requirements, including privacy preservation, anonymity, and untraceability should be guaranteed during communication. To realize digital twin technology, we need a secure and privacy-preserving communication scheme that can defend against these attacks and satisfy the security requirements mentioned above.

Blockchain technology can be a solution for guaranteeing data integrity and verifiability [7]–[10]. Once the data are stored in the blockchain, it is practically impossible to modify the recorded data, and users can easily verify the data using a Merkle hash tree. In the proposed model, we store the digital twin data on the cloud server and the data hash in the blockchain. Therefore, users can verify whether the data received from the cloud server have been modified. Furthermore, after the data are shared between the cloud server and the data user, the log transaction is uploaded to the blockchain. Data users can check whether their data are shared with authenticated data users through the log transactions. We also propose secure mutual authentication schemes between the data owner and cloud server, and the data owner and data user, suitable for the proposed system model. The network participants authenticate each other before communication to securely transmit digital twin data in wireless channels. The main contributions of this study are as follows.

- We establish a system model using cloud computing and blockchain for digital twin environments. The proposed model uses cloud computing for efficient digital twin data sharing, and it utilizes blockchain for data verifiability and integrity.
- We propose secure and privacy-preserving mutual authentication and key agreement schemes between a data owner and the cloud, and a data owner and data user to ensure secure communication in public channels. The proposed protocol can guarantee various security features including anonymity, untraceability, and perfect forward secrecy.
- We informally analyze the proposed scheme and formally analyze it using BAN logic and AVISPA to show the robustness of the proposed protocol. Furthermore, we compare the proposed protocol with other recently proposed schemes to demonstrate its efficiency and security.

A. STRUCTURE OF THE PAPER

In Section II, we introduce related work on digital twin environments and other communication schemes. In Sections III and IV, we present the proposed system model and propose a secure and verifiable digital twin data-sharing scheme, respectively. In Section V, we analyze the proposed scheme using informal and formal methods. In Section VI, we compare our scheme with other related schemes, and in Section VII, we conclude the paper.

II. RELATED WORK

We first introduce several papers that can help understand digital twin environments. In 2017, Alam and Saddik [11] proposed a digital twin architecture reference model for cloud-based cyber-physical systems (C2PS). In [11], physical systems are used to collect data from real-world assets and send them to the digital twins for simulation. Subsequently, depending on the simulation results, the digital twin sends control comments or notifies the physical system. Alam and Saddik proposed analytical models for C2PS and provided a telematics-based automotive driving assistance application following their proposed model. Liu *et al*. [12] proposed a cloud-based framework for healthcare services using digital twin technology. They mainly focused on combining digital twin technology and healthcare for elderly patients. In their scheme, health data are generated by medical devices, such as wristbands, portable electrocardiograms (ECGs), and radio frequency identification (RFID) cards, and the data are collected on PCs or smartphones. Subsequently, the collected data are transmitted to a remote cloud server through wireless channels such as Ethernet, mobile networks, and Wi-Fi. Liu *et al*. contributed to the construction of a conceptual model for cloud-based healthcare environments using a digital twin. Aheleroff *et al*. [13] developed a digital twin reference architecture and applied it to industrial cases. They focused on determining the digital twin reference architecture model in Industry 4.0 and introduced a digital twin as a server (DTaaS) in the reference model. Tao *et al*. proposed a four-layer reference architecture model of the physical, digital, cyber, and app layers. In the physical layer, data from physical objects are generated and transmitted to the digital layer, which records the data in raw

or different file formats. The cyber layer includes cloud processing and storage, and the app layer handles the applications.

In recent years, many attempts have been made to apply blockchain technology to digital twin environments. Wang *et al.* [14] proposed a sustainable digital twin management architecture for IoT environments. Wang *et al.* formalized a digital twin framework in IoT device-assisted services and focused on optimizing the delay of data transmission and controlling energy depletion to guarantee high data fidelity. They also utilized blockchain technology for network decentralization and efficient data sharing. Huang *et al.* [15] proposed a data management system for digital twins of products using blockchain. Their scheme utilizes blockchain for efficient and secure digital twin data sharing, storage, access, and authenticity. Shen *et al.* [16] proposed blockchain-based digital twin data sharing for smart manufacturing environments. They thoroughly classified the data-sharing scenarios in digital twin environments and presented a workflow for their blockchain-based secure digital twin data-sharing scheme. However, the scheme in [16] does not include a specific protocol, which is essential in wireless communication environments. These schemes [14]–[16] present methods for applying blockchain technology and digital twin data management systems. However, these schemes do not consider specific protocols for secure mutual authentication and data sharing in digital twin environments.

Further, we introduce several recent communication schemes proposed for IoT environments. Wu *et al.* [17] reported the security vulnerabilities of the scheme proposed by Chen *et al.* [18], and proposed an enhanced scheme for smart grid environments. Wu *et al.* analyzed their protocol using ProVerif and compared the computational and communication costs with those of other related schemes. Khatoon *et al.* [19] proposed a key agreement protocol between patients and servers in telecare medical information systems. They demonstrated that their scheme is more efficient than other schemes in the same environment, and that it can guarantee various security features. However, their scheme is vulnerable to known session-specific temporary information and cannot guarantee perfect forward secrecy [20]. Sengquata *et al.* proposed a two-factor authentication scheme for cyber physical systems. They reported that the scheme proposed by Sengupta *et al.* [21] was not robust to smart card stolen attacks and failed to protect user anonymity. Sengquata *et al.* also analyzed the security of their protocol and compared it with related schemes to demonstrate its superiority. Although these schemes [17], [19], [21] do not handle digital twin environments, they have recently been proposed as authentication schemes in similar environments.

III. SYSTEM MODEL

In this section, we introduce the network and threat models.

A. NETWORK MODEL

We present a network model of cloud-based digital twin environments using blockchain. The proposed model has four entities: a cloud server, data owner, data user, and blockchain. A detailed description of each entity is provided below.

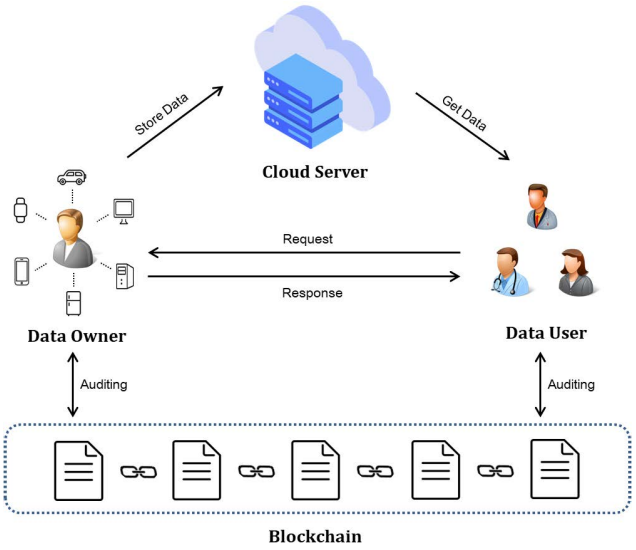

FIGURE 1. Proposed system model.

- **Cloud Server**: The cloud server has sufficient computation ability to simulate the digital twins and store the generated data. The cloud server receives data from physical assets of data owners after mutual key agreement. Then, the cloud server uses the data to conduct a simulation of digital twins in a virtual space. Thereafter, the cloud server shares the simulation results with the data owner. Furthermore, the cloud server can share the digital twin data with data users after receiving the data owner's confirmation. The cloud server also uploads hash values of the stored data and log data to the blockchain.
- **Data Owner**: Data owners have physical assets such as sensors and a wristband. Data owners collect data generated from the assets through their smartphone or device, and transmit the data to the cloud server after mutual authentication. Furthermore, when a data owner receives a data request from a data user, the data owner authorizes the cloud to share data with the data user. After the data sharing, the data owner can check the log record of the data sharing through the blockchain.
- **Data User**: Data users request digital twin data as required. Lab staff, researchers, and manufacturers can be considered as data users. After a data user is mutually authenticated with a data owner, the data user can access the digital data of the data owner stored on the cloud server. Then, the data user can obtain the required data and verify the data integrity through the blockchain.
- **Blockchain**: Blockchain stores hash values of the data stored on the cloud server and log records between the

cloud server and data users. The data hash values are used by data users to verify that the data are not modified, and the log records are used by data owners to check that their data are shared with legitimate data users. Each transaction contains a signature and is uploaded after the signature is verified by the smart contract.

B. THREAT MODEL

We apply the widely used "Dolev-Yao (DY) threat model" [23]–[25] to analyze the security of our protocol. The DY model assumes that adversary A has the following capabilities:

- A has complete control of the messages transmitted through wireless channels, and can eavesdrop on these messages or trace the messages.
- A can obtain a smart card of a legitimate user, and can extract the stored values of the smart card through the power analysis attack [26], [27].
- A has a dictionary of the identity and password, and can attempt to guess a legitimate user's identity and password simultaneously.
- An attacker can attempt security attacks such as session key disclosure, MITM, and impersonation attacks [28], [29].

IV. PROPOSED SCHEME

We propose secure authentication schemes for cloud-based digital twin data-sharing systems using blockchain. The proposed scheme involves initialization, registration, owner-cloud authentication, owner-user authentication, and data verification phases. TA initializes that network, and registers data owners, cloud servers, and data users. Then, data owners authenticates with cloud servers and transmits data generated from their physical assets in real-time. The cloud servers can perform simulation using digital twins of the assets, and the simulation results are generated, and the hash values of the generated data are stored in the blockchain. When a data user wants digital twin data, the data user authenticates to the data owner and can receive the data. After data is shared, the log record are stored in the blockchain. A detailed description of each phase is provided below.

A. INITIALIZATION

In the initialization phase, TA publishes public keys for the network. TA chooses a large prime p and two multiplicative groups G and G_T with order p, and generates $s_{TA} \in Z_p$ as a secret key of TA. TA also chooses $e : G \times G \rightarrow G_T$, $H(.) : \{0,1\}^* \rightarrow G$, $h(.) : \{0,1\}^* \rightarrow Z_p$, and a generator P of G, and computes $P_{TA} = s_{CA}.P$. TA publishes $(G, G_T, P, P_{TA}, p, H(.), h(.))$ and keeps s_{TA} secret.

B. REGISTRATION

Each entity should be registered with the TA to participate in the network. Before DO_i registration, TA generates ID_j and s_j and deploys them to CS_j, and $P_j = s_j.P$ and s_j are the public and secret keys of CS_j, respectively. DO_i can register with TA using its own smart device SD_i, which has sufficient capability to transmit the generated data to CS_j. DO_i chooses ID_i and PW_i, and generates $a_i \in Z_p$. Then, DO_i computes $HID_i = H(ID_i||PW_i||a_i)$ and sends $\{ID_i, HID_i\}$ to TA. TA first checks whether ID_i has already been registered. If not, TA generates s_i, and the fuzzy verifier $l \in \{2^5, 2^{10}\}$ computes $SID_i = s_i.HID_i$ and $P_i = s_i.P$ and then sends $\{SID_i, s_i, l\}$ to DO_i. After DO_i receives the message, it computes $HPW_i = h(ID_i||PW_i)$, $A_i = a_i \oplus HPW_i$, $B_i = s_i \oplus h(a_i||HPW_i)$, $C_i = SID_i \oplus h(s_i||a_i||HPW_i)$, and $Auth_i = h(s_i||a_i||SID_i)\ (mod\ l)$. DO_i stores $(A_i, B_i, C_i, Auth_i, l)$ in SD_i.

TABLE 1. Notations and their meanings.

Notation	Description
DO_i	i-th data owner
DU_k	k-th data user
CS_j	j-th cloud server
ID_i,PW_i	identity and password of DO_i
HID_i	pseudo identity of DO_i
SID_i	secret identity of DO_i
s_i, s_k	secret key of DO_i and DU_k
u_i, u_k	random numbers generated in session
Req	request message of DU_k
L_i, L_j	message digest of DO_i and CS_j
R_{ij}, R_{ji}	the shared diffie-hellman key between DO_i and CS_j
T_1, T_2, T_3, T_4	Timestamps
.	multiplication operation
$\|\|$	concatenation operation

C. OWNER-CLOUD AUTHENTICATION

DO_i should authenticate CS_j to send the data generated from their physical assets. DO_i inputs ID_i and PW_i to SD_i, then SD_i computes $HPW_i = h(ID_i||PW_i)$, $a_i = A_i \oplus HPW_i$, $s_i = B_i \oplus h(a_i||HPW_i)$, and $SID_i = C_i \oplus h(s_i||a_i||HPW_i)$, and checks $Auth_i \stackrel{?}{=} h(s_i||a_i||SID_i)\ modl$. If it is equal, SD_i generates r_i and T_1, and computes $HID_i = H(ID_i||PW_i||a_i)$, $R_i = r_i.a_i.P$, $R_{ij} = r_i.a_i.P_j$, $PID_i = HID_i \oplus h(R_{ij}||T_1)$, and $X_i = SID_i.h(HID_i||R_{ij}||T_1)$. Subsequently, DO_i transmits $\{R_i, PID_i, X_i, T_1\}$ to CS_j. After CS_j receives the message, CS_j computes $R_{ij} = s_j.R_i$ and $HID_i = PID_i \oplus h(R_{ij}||T_1)$, and checks $e(HID_i.h(HID_i||R_{ij}||T_1.P_{TA})) \stackrel{?}{=} e(X_i, P)$. If they are equal, CS_j generates r_j and T_2, and computes $R_j = r_j.P$, $R_{ji} = r_j.R_i$, $SK = h(R_{ij}||R_{ji}||HID_i)$, and $L_j = h(SK||R_{ij}||R_{ji})$. Then, CS_j sends $\{R_j||L_j||T_2\}$. DO_i receives the message, computes $R_{ji} = r_i.a_i.R_j$ and $SK = h(R_{ij}||R_{ji}||HID_i)$, and checks $L_j = h(SK||R_{ij}||R_{ji})$. Subsequently, the data generated from the physical assets of DO_i are transmitted to CS_j encrypted using SK, CS_j simulates using the data in the virtual space, and the results are transmitted to DO_i. Then, the hash value of the stored data in CS_j is generated and the signature of DO_i corresponding to the data is uploaded to the blockchain.

D. USER-OWNER AUTHENTICATION

DU_k can request DO_i digital twin data when required. DU_k generates a request message $Req_k \in Z_p$ and chooses a random number u_k and timestamp T_3. Then,

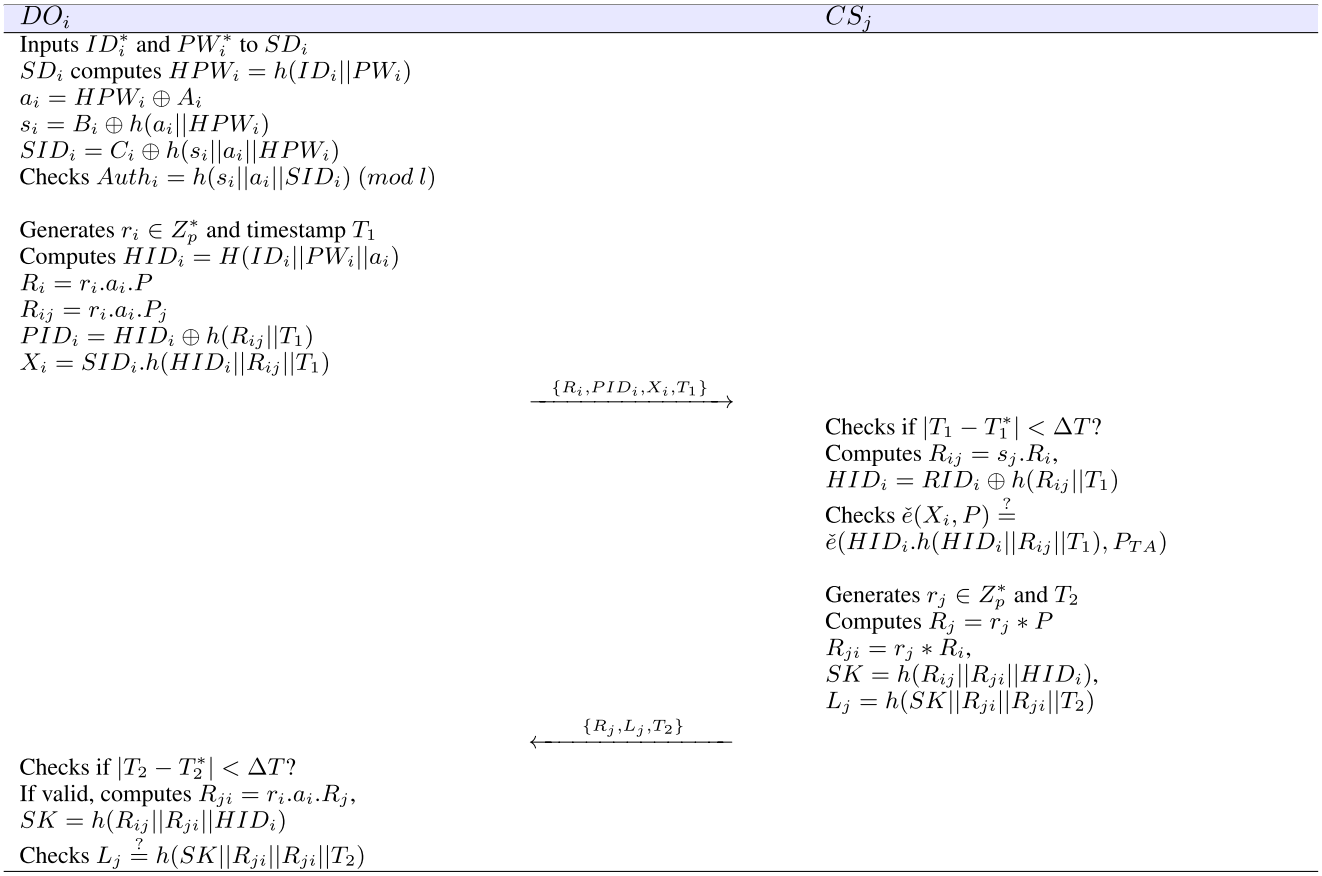

FIGURE 2. Authentication phase between DO_i and CS_j.

DU_k computes $u_k.a_k.P = U_k$, $u_k.a_k.P_i = U_{ki}$ $PID_k = HID_k \oplus h(U_{ki}||T_3)$, $M_k = Req_k \oplus h(HID_k||U_{ki}||T_3)$, and $X_k = SID_k.h(HID_k||Req_k||U_{ki}||T_1)$, and sends $\{U_k, PID_k, M_k, X_k, T_3\}$. During this phase, DO_i authenticates DU_k and sends a response message to DU_k. DO_i receives the message from DU_k, computes $U_{ki} = s_k.U_k$, $HID_k = PID_k \oplus h(U_{ki}||T_3)$, and $Req_k = M_k \oplus h(HID_k||U_{ki}||T_3)$, and verifies $e(HID_k.h(HID_k||Req_k||U_{ki}||T_3), P_{TA}) \stackrel{?}{=} e(X_k, P)$. If they are equal, DO_i generates a random number u_i and computes $U_i = u_i.P$, $U_{ik} = u_i.X_k$, $SK_{ik} = h(U_{ki}||U_{ik}||HID_k||HID_i)$, and $L_i = h(SD_{ik}||U_{ki}||U_{ik})$. Then, DO_i sends $\{U_i, L_i, T_4\}$ to DU_k, and DU_k computes $U_{ik} = u_k.a_k.U_i$, $SK_{ik} = h(U_{ki}||U_{ik}||HID_k||HID_i)$, and checks $L_i = h(SK_{ik}||U_{ki}||U_{ik})$. Finally, the shared secret key SK_{ik} is agreed upon between DO_i and DU_k. Then, DO_i sends SK_{ik} to CS_j, and CS_j encrypts the data of DO_i using the key and sends the encrypted data to DU_k. DU_k can decrypt it and verify data integrity through the hash values stored in the blockchain.

V. SECURITY ANALYSIS

In this section, we demonstrate that the proposed protocol defeats a variety of attacks using informal analysis, and we implement formal analysis using the "Burrows–Abadi–Needham (BAN) logic" [22] and "Automated Validation of Internet Security Protocols and Applications (AVISPA)" software validation tool [30]. We mainly analyze the user-owner authentication phase because both authentication schemes can be analyzed similarly, and user-owner authentication schemes require higher security compared to the owner-cloud authentication scheme.

A. INFORMAL ANALYSIS

We informally demonstrate that the proposed protocol is secure against a variety of attacks.

1) REPLAY AND MITM ATTACKS

An adversary A can attempt a replay attack using the messages $\{U_k, PID_k, M_k, X_k, T_3\}$ to analyze the received messages $\{U_i, L_i, T_2\}$ to trace message senders or attack the network. However, A cannot obtain any information without knowing the random values u_k and u_i or secret keys s_k and s_i, and cannot trace the message senders. Furthermore, A also fails to perform an MITM attack because A cannot know HID_k and SID_k, which are encrypted using U_{ki}. Therefore, the proposed scheme is secure against replay and MITM attacks.

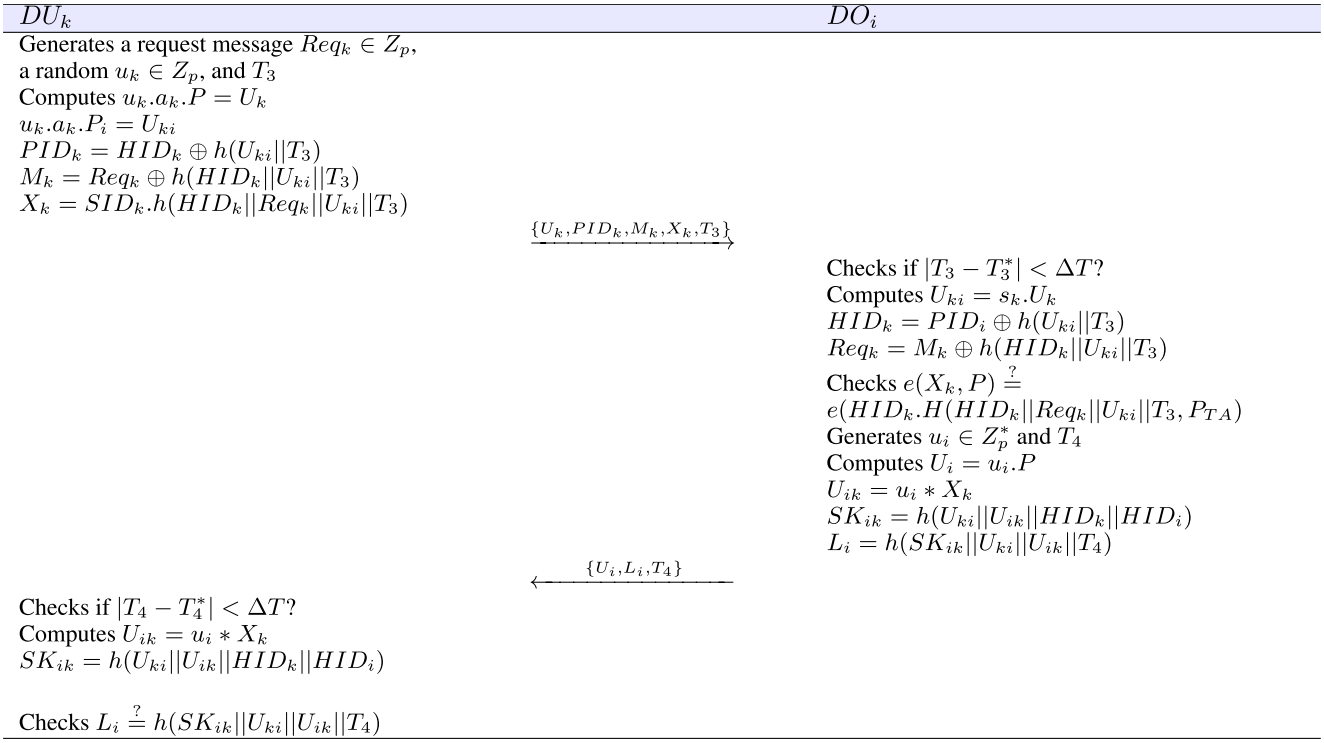

FIGURE 3. Authentication phase between DU_k and DO_i.

2) SMART DEVICE STOLEN ATTACK

Under this attack, A can obtain SD_i and the stored values $(A_i, B_i, C_i, Auth_i, l)$. However, A cannot obtain any information about DO_i because secret values such as a_i and s_i are masked by ID_i and PW_i, which are veiled. Therefore, A cannot acquire any meaningful information from SD_i, and the proposed protocol is robust to smart device stolen attacks.

3) OFFLINE GUESSING ATTACK

A can attempt to guess ID_i and PW_i using the messages obtained from public channels and extracted values from SD_i. Then, A has $\{U_k, PID_k, M_k, X_k, T_3\}$ and $\{U_i, L_i, T_4\}$. A must guess ID_i and PW_i using SD_i because the messages from the public channels do not contain information about ID_i or PW_i. However, if A successfully logs in using guessed ID_i' and PW_i^*, A cannot be convinced that the guessed values are correct because $Auth_i$ is masked with fuzzy verifier l. Therefore, the proposed protocol is secure against offline guessing attacks.

4) IMPERSONATION ATTACK

This attack is a case that A impersonate a legitimate user in the network and successfully authenticates with another entity. In our scheme, A can impersonate a legitimate DO_i or DU_k. To impersonate a DO_i or DU_k, A should acquire (HID_i, SID_i) or (HID_k, SID_k) because these values are used to authenticate other entities. However, A cannot obtain these values, and the proposed protocol is therefore secure against impersonation attacks.

5) SESSION KEY DISCLOSURE ATTACK

It is an attack that considers the case where A calculate session key directly. A can attempt to directly calculate the session key using the values obtained from public channels and SD_i. A must acquire U_{ki}, U_{ik}, HID_k, and HID_i to calculate session key SK_{ik}. However, A cannot know any of these values because they are masked using session random numbers u_i and u_k, as well as secret keys a_k and s_i. Therefore, the proposed protocol is robust to session key disclosure attacks.

6) PERFECT FORWARD SECRECY

It is a security feature that checks the security of session key when network long-term keys are leaked. When the network is compromised, all long-term values s_i, a_i, s_k, a_k, SID_i, and SID_k can be leaked to A. However, A still cannot calculate SK_{ik} because it contains session random numbers u_i and u_k, which are deleted after the session is completed. Thus, the proposed protocol guarantees perfect forward secrecy.

7) PRIVILEGED-INSIDER ATTACK

A may be a privileged insider belonging to TA. Then, A can obtain the secret values that are used for the registration of DO_i, and can attempt to use them in other networks where DO_i is registered. However, sensitive information such as PW_i is not transmitted during the registration process, and the proposed protocol is secure against privileged-insider attacks.

8) KNOWN SESSION-SPECIFIC TEMPORARY INFORMATION (KSSTI) ATTACK

It is an attack that assumes session random numbers are leaked and check the security of the session key. Under the KSSTI attack, we assume that the session random numbers u_i and u_k are leaked, and we check whether the session key SK_{ik} is secure. SK_{ik} is calculated using U_{ik}, U_{ki}, HID_k, and HID_i. However, although A can compute $U_{ik} = u_i.X_k$, A cannot compute U_{ki}, and the proposed protocol is therefore secure against KSSTI attacks.

9) ANONYMITY AND UNTRACEABILITY

A can trace DO_i or DU_k using messages transmitted over public channels. However, the pseudo-identities HID_i and HID_k are not disclosed in the transmitted messages, and A cannot know who sent the message. Therefore, the proposed protocol guarantees the anonymity and untraceability of DO_i and DU_k.

10) DATA VERIFIABILITY

The proposed scheme can guarantee data verifiability by using blockchain. After DU_k receives the requested data from CS_j, DU_k can verify the data integrity through the hash value stored in the blockchain. If it is not equal, DU_k can perceive that the data were modified, and can determine that the data are invalid.

B. BAN LOGIC ANALYSIS

In this section, we present a BAN logic of the proposed protocol. BAN logic is a formal analysis method that can verify the correctness of an authentication protocol. Table 2 lists the notations and their meanings, and the following are basic rules used for the BAN logic analysis.

TABLE 2. BAN logic notations.

Notation	Description
p_1, p_2	Two principals
s_1, s_2	Two statements
SK	The session key
$p_1 \mid\equiv s_1$	p_1 **believes** s_1
$p_1 \mid\sim s_1$	p_1 once **said** s_1
$p_1 \Rightarrow s_1$	p_1 **controls** s_1
$p_1 \lhd s_1$	p_1 **receives** s_1
$\#s_1$	s_1 is **fresh**
$(s_1)_K$	s_1 is **encrypted** with K
$p_1 \overset{K}{\longleftrightarrow} p_2$	p_1 and p_2 have **shared** key K

1. Message meaning rule (MMR):

$$\frac{p_1 \mid\equiv p_1 \overset{K}{\leftrightarrow} p_2, \quad p_1 \lhd (s_1)_K}{p_1 \mid\equiv p_2 \mid\sim s_1}$$

2. Nonce verification rule (NVR):

$$\frac{p_1 \mid\equiv \#(s_1), \quad p_1 \mid\equiv p_2 \mid\sim s_1}{p_1 \mid\equiv p_2 \mid\equiv s_1}$$

3. Jurisdiction rule (JR):

$$\frac{p_1 \mid\equiv p_2 \mid\Longrightarrow s_1, \quad p_1 \mid\equiv p_2 \mid\equiv s_1}{p_1 \mid\equiv s_1}$$

4. Belief rule (BR):

$$\frac{p_1 \mid\equiv (s_1, s_2)}{p_1 \mid\equiv s_1}$$

5. Freshness rule (FR):

$$\frac{p_1 \mid\equiv \#(s_1)}{p_1 \mid\equiv \#(s_1, s_2)}$$

1) GOALS

The goals of proving the correctness of our protocol are defined as follows:

Goal 1: $DU_k \mid\equiv DU_k \overset{SK}{\longleftrightarrow} DO_i$
Goal 2: $DU_k \mid\equiv DO_i \mid\equiv DU_k \overset{SK}{\longleftrightarrow} DO_i$
Goal 3: $DO_i \mid\equiv DU_k \overset{SK}{\longleftrightarrow} DO_i$
Goal 4: $DO_i \mid\equiv DU_k \mid\equiv DU_k \overset{SK}{\longleftrightarrow} DO_i$

2) ASSUMPTIONS

The BAN logic assumptions of our protocol are as follows:

A_1: $DO_i \mid\equiv \#(T_3)$
A_2: $DU_k \mid\equiv \#(T_4)$
A_3: $DU_k \mid\equiv DO_i \Rightarrow (DU_k \overset{SK}{\longleftrightarrow} DO_i)$
A_4: $DO_i \mid\equiv DU_k \Rightarrow (DU_k \overset{SK}{\longleftrightarrow} DO_i)$
A_5: $DU_k \mid\equiv DU_k \overset{U_{ki}}{\longleftrightarrow} DO_i$
A_6: $DO_i \mid\equiv DU_k \overset{U_{ki}}{\longleftrightarrow} DO_i$

3) IDEALIZED FORMS

Based on the BAN logic, the idealized forms of our protocol can be described as follows:

$$Msg_1 \ : \ DU_k \rightarrow DO_i : (U_k, HID_k, T_3)_{U_{ki}}$$

$$Msg_2 \ : \ DO_i \rightarrow DU_k : (U_i, HID_i, T_4)_{U_{ki}}$$

4) BAN LOGIC PROOF

We implemented the BAN logic analysis of our protocol as follows:

Step 1: DO_i receives Msg_1.

$$S_1 : DO_i \lhd (U_k, HID_k, T_3)_{U_{ki}}$$

Step 2: We can apply MMR using Msg_1 and A_5.

$$S_2 : DO_i \mid\equiv DU_k \mid\sim (U_k, HID_k, T_3)$$

Step 3: We can apply FR using Msg_1 and A_1.

$$S_3 : DO_i \mid\equiv \#(U_k, HID_k, T_3)$$

Step 4: We can apply NVR using S_2 and S_3.

$$S_4 : DO_i| \equiv DU_k| \equiv (U_k, HID_k, T_3)$$

Step 5: We can apply BR using S_4.

$$S_5 : DO_i| \equiv DU_k| \equiv (HID_k)$$

Step 6: DO_i can calculate the session key $SK = h(HID_k||HID_i||U_{ik}||U_{ki})$.

$$S_6 : DO_i| \equiv DU_k| \equiv (DU_k \overset{SK}{\longleftrightarrow} DO_i) \quad \textbf{(Goal4)}$$

Step 7: We can apply JR using A_4 and S_6.

$$S_7 : DO_i| \equiv (DU_k \overset{SK}{\longleftrightarrow} DO_i) \quad \textbf{(Goal3)}$$

Step 8: DU_k receives Msg_2

$$S_8 : DU_K \lhd (U_i, HID_i, T_4)_{U_{ki}}$$

Step 9: We can apply MMR using Msg_1 and A_6

$$S_9 : DU_k| \equiv DO_i| \sim (U_i, HID_i, T_4)$$

Step 10: We can apply FR using Msg_2 and A_2

$$S_{10} : DU_k| \equiv \#(U_i, HID_i, T_4)$$

Step 11: We can apply NVR using S_9 and S_{10}.

$$S_{11} : DU_k| \equiv DO_i| \equiv (U_i, HID_i, T_4)$$

Step 12: We can apply BR using S_{11}.

$$S_{12} : DU_k| \equiv DO_i| \equiv (HID_i)$$

Step 13: DU_i can calculate the session key $SK = h(HID_k||HID_i||U_{ik}||U_{ki})$.

$$S_{13} : DU_k| \equiv DO_i| \equiv (DU_k \overset{SK}{\longleftrightarrow} DO_i) \quad \textbf{(Goal2)}$$

Step 14: We can apply JR using A_3 and S_{13}.

$$S_{14} : DU_k| \equiv (DU_k \overset{SK}{\longleftrightarrow} DO_i) \quad \textbf{(Goal1)}$$

C. AVISPA SIMULATION

We performed formal analysis using the widely accepted "Automated Validation of Internet Security Protocols and Applications (AVISPA)" simulation tool [30]–[32], which can verify that an authentication protocol is robust against replay and MITM attacks. The AVISPA tool implements a communication protocol through the High-Level Protocol Specification Language (HLPSL) [33], and the HLPSL has four back-end models, which include "On-the-Fly Model Checker (OFMC)" [34], "Tree Automata based on Automatic Approximations for Analysis of Security Protocols (TA4SP)", "Constraint Logic-based Attack Searcher (CL-AtSe)" [35], and "SAT-based Model Checker (SATMC)". When the code is input as one of these models, it is converted to "Intermediate Format (IF)", and the output is in "Output Format (OF)". The AVISPA tool uses two models, OFMC and CL-AtSe, for formal verification because these two models support exclusive-OR operations. If an authentication protocol is safe under these two models, the simulated protocol can be considered to be secure against replay and MITM attacks. Figure 4 shows the role of the data owner implemented in the AVISPA simulation. The implementation details of the data owner and TA are similar to those of the data user. Figure 5 shows the roles of the session, goals, and environment. The simulation results are shown in Figure 6, and we can deduce that the proposed protocol is robust to replay and MITM attacks.

```
role user(DO,TA,DU:agent, SKtai,SKtak:symmetric_key,H:hash_func, SND,RCV:channel(dy))
played_by DU
def=
local State: nat,
MUL: hash_func,
P, IDi,IDk, HIDi,HIDk,SIDi,SIDk,Sta,Ai,Sk,PIDk,Req,Uki,Uik,Ui,Uii,Uk,Ukk,Li,T1,T2,Xk,Mk,Ak,SKik,Si,PWi,PWk,Pk,Pi:text
const ta_sec1,ta_sec2,do_sec1,do_sec2,do_sec3,do_du_1,do_du_2,du_do_1,du_do_2,du_sec1,du_sec2,du_sec3:protocol_id
init State := 2
transition
1. State = 2 /\ RCV(start) =|>
State' := 3 /\ Ak' := new()
          /\ HIDk' := H(IDk.PWk.Ak')
          /\ SND{{IDk.HIDk'}_SKtak}
          /\ secret(IDk,du_sec1,{DU,TA})
          /\ secret({PWk,Ak'},du_sec2,{DU})
1. State = 3 /\ RCV({MUL(H(IDk.PWk.Ak').Sta).Sk'}_SKtak) =|>
State' := 4/\ HIDk' := H(IDk.PWk.Ak')
          /\ SIDk' := MUL(H(IDk.PWk.Ak').Sta)
          /\ SND{{IDk.HIDk'}_SKtak}
          /\ secret(IDk,du_sec3,{DU,TA})
2. State = 4 /\ RCV(start) =|>
  State' := 5 /\ HIDk' := MUL(H(IDk.PWk.Ak).P)
          /\ Uk' :=new() /\ T1':=new()
          /\ Ukk' :=MUL(Uk'.Ak.P)
          /\ Uki' :=MUL(Uk'.Ak.Pi)
          /\ PIDk' := xor(HIDk',H(Uki'.T1'))
          /\ Mk' := xor(Req,H(HIDk'.Uki'.T1'))
          /\ Xk' := MUL{MUL(H(IDk.PWk.Ak).Sta).H(HIDk'.Req.Uki'.T1')}
          /\ SND{{Ukk',PIDk',Mk',Xk',T1'}}
          /\ witness(DU,DO,du_do_1,Uki')
3. State = 5 /\ RCV(MUL(Ui'.P).H(H(MUL(Sk.Uk') .MUL(Ui'.MUL(MUL(H(IDk.PWk.Ak').Sta).H(H(IDk.PWk.Ak').Req.MUL(Uk'.Ak'.Pk).T1'))).
H(H(IDk.PWk.Ak').H(ID.PWi.Ai')).MUL(Sk.Uk').MUL(Ui'.MUL(MUL(H(IDk.PWk.Ak').Sta).H(H(IDk.PWk.Ak').Req.MUL(Uk'.Ak'.Pk).T1'))))).T2') =|>
State' := 6 /\ Uik' := MUL(Uk'.Ak'.MUL(Ui'.P))
          /\ SKik' := H(MUL(Sk.Uk').Uik'.MUL(H(IDk.PWk.Ak').P).MUL(H(IDk.PWk.Ak).P))
          /\ Li' := H(SKik'.H(MUL(Sk.Uk').Uik'))
          /\ request(DU,DO,du_do_2,SKik')
end role
```

FIGURE 4. *Role* of data user.

```
role session(DO,TA,DU:agent, SKtai,SKtak:symmetric_key,H:hash_func)
def=
local SN1, SN2, SN3, RV1, RV2, RV3: channel(dy)
composition
owner(DO,TA,DU, SKtai,SKtak, H, SN1, RV1)
/\ user(DO,TA,DU, SKtai,SKtak, H, SN2, RV2)
/\ authority(DO,TA,DU, SKtai,SKtak, H, SN3, RV3)
end role

role environment()
def=
const ta,du,do: agent,
sktai,sktak: symmetric_key,
h,mul: hash_func,
idi,idk,p,pi,pk:text,
ta_sec1,ta_sec2,do_sec1,do_sec2,do_sec3,do_du_1,do_du_2,du_do_1,du_do_2,du_sec1,du_sec2,du_sec3: protocol_id
intruder_knowledge = {idi,idk,p,pi,pk, h, mul}
composition
session(ta,du,do,sktai,sktak,h)
/\session(i,du,do,sktai,sktak,h)
/\session(ta,i,do,sktai,sktak,h)
/\session(ta,du,i,sktai,sktak,h)
end role

goal
secrecy_of ta_sec1,ta_sec2,do_sec1,do_sec2,do_sec3,du_sec1,du_sec2,du_sec3
authentication_on do_du_1,do_du_2,du_do_1,du_do_2
end goal

environment()
```

FIGURE 5. *Role* of session, goals, and environment.

VI. PERFORMANCE ANALYSIS

We compared the proposed authentication protocol with recently proposed schemes in similar environments [17], [19], [21].

A. COMPUTATIONAL COST

We refer to the experiments performed in [17], which were implemented using the Java pairing-based cryptography library. The experiment was performed on a computer with a 2.3 GHz Intel it-8300H quad core processor and 16 GB memory, and the time cost of each operation was as follows:

- T_{bp}: The computational cost of the bilinear pairing operation $\approx$ 17.4 ms

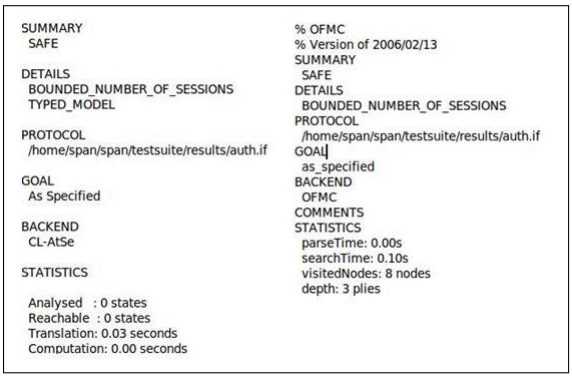

FIGURE 6. Simulation results.

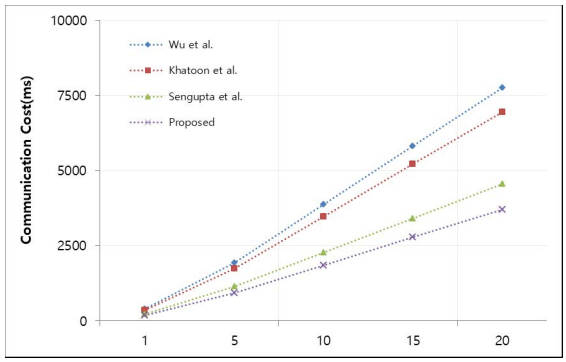

FIGURE 7. Computational cost as the number of authentications increases.

- T_{hp}: The computational cost of the map-to-point hash operation $\approx$ 42.1 ms
- T_{exp}: The computational cost of the modular exponentiation operation $\approx$ 15.8 ms
- T_{mul}: The computational cost of the point scalar multiplication operation $\approx$ 13.5 ms
- T_{add}: The computational cost of the point addition operation $\approx$ 0.48 ms

TABLE 3. Computational cost comparison.

Scheme	Total execution time
Wu *et al.* [17]	387.46 ms
Khatoon *et al.* [19]	347.1 ms
Sengupta *et al.* [21]	227.48 ms
Proposed	184.9 ms

We neglect the computational cost of an exclusive-OR operation and a one-way hash function because they require relatively little computational cost. We analyzed the time cost of the login and authentication phases. First, the total computational cost of the scheme proposed in [17] is $2T_{bp} + 11T_{mul} + 4T_{hp} + 2T_{add} \approx 387.46$ ms. Second, the total computational cost of the scheme proposed in [19] is $4T_{hp} + 2T_{bp} + 7T_{mul} \approx 347.1$ ms. Next, the total computational cost of the scheme proposed in [21] is $4T_{bp} + 4T_{mul} + 2T_{hp} + 41T_{add} \approx 227.48$ ms. In addition, the proposed scheme incurs a $2T_{bp} + T_{hp} + 8T_{mul} \approx 193.76$ ms time cost. A comparison of the computational cost is summarized in Table 3, and the computational cost increases according to the number of authentications, as shown in Figure 7. As represented in Table 3 and Figure 7, the proposed protocol has a lower computational cost compared to existing schemes [17], [19], [21], and we can deduce that the proposed scheme is more efficient than the existing schemes.

B. COMMUNICATION COST

We analyzed the communication costs of our proposed protocol and other protocols [17], [19], [21]. We set the bit sizes of an identity, a random number, a timestamp, a one-way cryptographic hash output, and the group elements of G_1 as 128 bits, 160 bits, 32 bits, 256 bits, and 1024 bits, respectively.

TABLE 4. Communication cost comparison.

Scheme	Total communication cost
Wu *et al.* [17]	5664 bits
Khatoon *et al.* [19]	2624 bits
Sengupta *et al.* [21]	5184 bits
Proposed	4672 bits

In the scheme in [17], the first message is $(Id_i, R_{in}, R_{i1}, R_{si}, T_1)$ and the second message is $(Id_j, R_{jn}, R_{j1}, h_j)$, which include five group elements of G_1, a hash output, two identities, and a timestamp. The total communication cost is $5120 + 256 + 256 + 32 = 5664$ bits. In the scheme in [19], the first message is $(R_i, T_i, Auth_i)$ and the second message is $(R_s, T_s, Auth_s)$. These messages include two group elements of G_1, two hash outputs, and two timestamps, and the scheme incurs a total cost of $2048 + 512 + 64 = 2624$ bits. In the scheme in [21], the first message is $(CID_i, N_i, C_i, F_i, T_i)$ and the second message is (a, T_{ss}). These messages include five group elements of G_1 and two timestamps, and the total number of communication bits is $5120 + 64 = 5184$ bits. The first message of the proposed scheme is $(U_k, SID_k, M_k, X_k, T_3)$, and the second message is (U_i, L_i, T_4). These include four group elements, two hash outputs, and two timestamps, and the total communication cost is $4096 + 512 + 64 = 4672$ bits. Although the proposed scheme has a slightly higher communication cost compared to the scheme proposed in [19], it is more efficient than other schemes [17], [21], and as we present in VI-C, our proposed protocol has superior security compared to the related schemes.

C. SECURITY FEATURES

We compared the security features of our method with those of related protocols [17], [19], [21]. We considered a) "resistance to replay and MITM attacks," b) "resistance to session key disclosure attack," c) "resistance to offline guessing attack," d) "resistance to impersonation attack," e) "preservation of perfect forward secrecy," f) "resistance to privileged-insider attack," g) "resistance to KSSTI

TABLE 5. Comparison of security features.

Security features	[17]	[19]	[21]	Proposed
Replay and MITM attacks	O	O	O	O
Session key disclosure attack	O	O	O	O
Offline guessing attack	O	O	O	O
Impersonation attack	O	O	O	O
Perfect forward secrecy	O	X	O	O
Privileged-insider attack	O	O	O	O
KSSTI attack	X	X	–	O
Anonymity and untraceability	X	O	O	O
Data verifiability	X	X	X	O

X: Insecure, O: Secure, –: Not considered.

attack," h) "preservation of anonymity and untraceability," and i) "data verifiability." The comparison results are summarized in Table 5. As shown in Table 5, it is clear that the proposed scheme has superior security compared to other related schemes in similar environments [17], [19], [21].

VII. CONCLUSION

In this study, we presented a system model for a cloud-based digital twin environment using blockchain, and we proposed a privacy-preserving communication scheme for the proposed model. In the proposed scheme, data generated from physical assets are collected by the smart device of a data owner, and the smart device transmits the data to the cloud server after mutual authentication. Then, the data can be transmitted in real time, and the simulation data can also be transmitted to the data owner in a secure manner. In addition, when a data user requests the data from the data owner, the data owner authenticates the data user and can share the requested data through the cloud server. Then, the data user can receive the data and verify the data integrity through the blockchain. By performing informal analyses, we demonstrated that the proposed scheme is robust to various types of attacks. Furthermore, we demonstrated the robustness of our scheme using BAN logic and proved that the proposed model is resistant to replay and MITM attacks using AVISPA simulation. We also compared the performance of our scheme with those of other schemes proposed for similar environments, and showed that the proposed scheme is more robust and efficient. Consequently, the proposed scheme can be used in the digital twin environment. In the future, we plan to implement the proposed scheme and propose an enhanced version.

REFERENCES

[1] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary Perspectives on Complex Systems*. Berlin, Germany: Springer, 2017.
[2] B. Piascik, J. Vickers, D. Lowry, S. Scotti, J. Stewart, and A. Calomino, "Materials, structures, mechanical systems, and manufacturing roadmap," NASA TA12, Washington, DC, USA, Tech. Rep., 2012, pp. 12–22.
[3] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE Access*, vol. 7, pp. 167653–167671, 2019.
[4] C. K. Lo, C. H. Chen, and R. Y. Zhong, "A review of digital twin in product design and development," *Adv. Eng. Informat.*, vol. 48, Apr. 2021, Art. no. 101297.
[5] B. Putz, M. Dietz, P. Empl, and G. Pernul, "EtherTwin: Blockchain-based secure digital twin information management," *Inf. Process. Manage.*, vol. 58, no. 1, Jan. 2021, Art. no. 102425.
[6] M. Liu, S. Fang, H. Dong, and C. Xu, "Review of digital twin about concepts, technologies, and industrial applications," *J. Manuf. Syst.*, vol. 58, pp. 346–361, Jan. 2021.
[7] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu, "Blockchain based data integrity service framework for IoT data," in *Proc. IEEE Int. Conf. Web Services*, Honolulu, HI, USA, Jun. 2017, pp. 468–475.
[8] P. Wei, D. Wang, Y. Zhao, S. K. S. Tyagi, and N. Kumar, "Blockchain data-based cloud data integrity protection mechanism," *Future Gener. Comput. Syst.*, vol. 102, pp. 902–911, Jan. 2020.
[9] H. Wang and J. Zhang, "Blockchain based data integrity verification for large-scale IoT data," *IEEE Access*, vol. 7, pp. 164996–165006, 2019.
[10] S. Nakamoto. (2008). *Bitcoin: A Peer-To-Peer Electronic Cash System.* Accessed: Jul. 2020. [Online]. Available: http://bitcoin.org/bitcoin.pdf
[11] K. M. Alam and A. El Saddik, "C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE Access*, vol. 5, pp. 2050–2062, 2017.
[12] Y. Liu, L. Zhang, Y. Yang, L. Zhou, L. Ren, F. Wang, R. Liu, Z. Pang, and M. J. Deen, "A novel cloud-based framework for the elderly healthcare services using digital twin," *IEEE Access*, vol. 7, pp. 49088–49101, 2019.
[13] S. Aheleroff, X. Xu, R. Y. Zhong, and Y. Lu, "Digital twin as a service (DTaaS) in Industry 4.0: An architecture reference model," *Adv. Eng. Informat.*, vol. 47, Jan. 2021, Art. no. 101225.
[14] C. Wang, Z. Cai, and Y. Li, "Sustainable blockchain-based digital twin management architecture for IoT devices," *IEEE Internet Things J.*, early access, Feb. 23, 2022, doi: 10.1109/JIOT.2022.3153653.
[15] S. Huang, G. Wang, Y. Yan, and X. Fang, "Blockchain-based data management for digital twin of product," *J. Manuf. Syst.*, vol. 54, pp. 361–371, Jan. 2020.
[16] W. Shen, T. Hu, C. Zhang, and S. Ma, "Secure sharing of big digital twin data for smart manufacturing based on blockchain," *J. Manuf. Syst.*, vol. 61, pp. 338–350, Oct. 2021.
[17] T.-Y. Wu, Y.-Q. Lee, C.-M. Chen, Y. Tian, and N. A. Al-Nabhan, "An enhanced pairing-based authentication scheme for smart grid communications," *J. Ambient Intell. Humanized Comput.*, early access, pp. 1–13, Jan. 2021. [Online]. Available: https://link.springer.com/article/10.1007/s12652-020-02740-2
[18] Y. Chen, J. Martinez, P. Castillejo, and L. López, "A bilinear map pairing based authentication scheme for smart grid communications: PAuth," *IEEE Access*, vol. 7, pp. 22633–22643, 2019.
[19] S. Khatoon, S. M. M. Rahman, M. Alrubaian, and A. Alamri, "Privacy-preserved, provable secure, mutually authenticated key agreement protocol for healthcare in a smart city environment," *IEEE Access*, vol. 7, pp. 47962–47971, 2019.
[20] M. Nikooghadam and H. Amintoosi, "Cryptanalysis of Khatoon *et al.*'s ECC-based authentication protocol for healthcare systems," 2019, *arXiv:1906.08424*.
[21] A. Sengupta, A. Singh, P. Kumar, and T. Dhar, "A secure and improved two factor authentication scheme using elliptic curve and bilinear pairing for cyber physical systems," *Multimedia Tools Appl.*, vol. 81, pp. 1–24, May 2022.
[22] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, Feb. 1990.
[23] S. Son, J. Lee, Y. Park, Y. Park, and A. K. Das, "Design of blockchain-based lightweight V2I handover authentication protocol for VANET," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1346–1358, May 2022.
[24] J. Lee, S. Yu, M. Kim, Y. Park, and A. K. Das, "On the design of secure and efficient three-factor authentication protocol using honey list for wireless sensor networks," *IEEE Access*, vol. 8, pp. 107046–107062, 2020.
[25] P. Bagga, A. K. Das, M. Wazid, J. J. P. C. Rodrigues, K.-K.-R. Choo, and Y. Park, "On the design of mutual authentication and key agreement protocol in internet of vehicles-enabled intelligent transportation system," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1736–1751, Feb. 2021.
[26] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO*. Berlin, Germany: Springer, 1999, pp. 388–397.
[27] S. Yu, J. Lee, Y. Park, Y. Park, S. Lee, and B. Chung, "A secure and efficient three-factor authentication protocol in global mobility networks," *Appl. Sci.*, vol. 10, no. 10, p. 3565, May 2020.
[28] J. Ryu, J. Oh, D. Kwon, S. Son, J. Lee, Y. Park, and Y. Park, "Secure ECC-based three-factor mutual authentication protocol for telecare medical information system," *IEEE Access*, vol. 10, pp. 11511–11526, 2022.
[29] D. K. Kwon, S. J. Yu, J. Y. Lee, S. H. Son, and Y. H. Park, "WSN-SLAP: Secure and lightweight mutual authentication protocol for wireless sensor networks," *Sensors*, vol. 21, no. 3, p. 936, Jan. 2021.

[30] L. Viganò, "Automated security protocol analysis with the AVISPA tool," *Electron. Notes Theor. Comput. Sci.*, vol. 155, pp. 61–86, May 2006.
[31] S. Yu and Y. Park, "A robust authentication protocol for wireless medical sensor networks using blockchain and physically unclonable functions," *IEEE Internet Things J.*, early access, May 2, 2022, doi: 10.1109/JIOT.2022.3171791.
[32] K. Park, J. Lee, A. K. Das, and Y. Park, "BPPS: Blockchain-enabled privacy-preserving scheme for demand-response management in smart grid environments," *IEEE Trans. Dependable Secure Comput.*, early access, Mar. 29, 2022, doi: 10.1109/TDSC.2022.3163138.
[33] D. Von Oheimb, "The high-level protocol specification language HLPSL developed in the EU project AVISPA," in *Proc. 3rd APPSEM II Workshop Appl. Semantics (APPSEM)*, Frauenchiemsee, Germany, 2005, pp. 1–17.
[34] D. Basin, S. Mödersheim, and L. Viganò, "OFMC: A symbolic model checker for security protocols," *Int. J. Inf. Secur.*, vol. 4, no. 3, pp. 181–208, 2005.
[35] M. Turuani, "The CL-Atse protocol analyser," in *Proc. Int. Conf. Rewriting Techn. Appl.*, Seattle, WA, USA, Aug. 2006, pp. 227–286.

SEUNGHWAN SON received the B.S. degree in mathematics from Kyungpook National University, Daegu, South Korea, in 2019, where he is currently pursuing the M.S. degree with the School of Electronic and Electrical Engineering. His research interests include authentication, blockchain, cryptography, and information security.

DEOKKYU KWON received the B.S. degree in electronics engineering and the M.S. degree in electronics and electrical engineering from Kyungpook National University, Daegu, South Korea, in 2020 and 2022, respectively. He is currently pursuing the Ph.D. degree with the School of Electronic and Electrical Engineering. His research interests include the Internet of Drones, wireless sensor networks, mutual authentication, and information security.

JOONYOUNG LEE (Student Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2018 and 2020, respectively, where he is currently pursuing the Ph.D. degree with the School of Electronics Engineering. His research interests include authentication, the Internet of Things, and information security.

SUNGJIN YU (Student Member, IEEE) received the B.S. degree in electronics engineering from Daegu University, in 2017, and the M.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2019. He is currently pursuing the Ph.D. degree in electronics and electrical engineering. He is a Researcher with the Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea. His research interests include blockchain, authentication, information security, VANET, FANET, the Internet of Vehicles, and the Internet of Drones.

NAM-SU JHO received the B.S. degree in mathematics from the Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 1999, and the Ph.D. degree in mathematics from Seoul National University, South Korea, in 2007. Since 2007, he has been with the Electronics and Telecommunications Research Institute (ETRI), Daejeon, as a Principal Researcher. His research interests include cryptography and information theory.

YOUNGHO PARK (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electronics engineering from Kyungpook National University, Daegu, South Korea, in 1989, 1991, and 1995, respectively. From 1996 to 2008, he was a Professor at the School of Electronics and Electrical Engineering, Sangju National University, South Korea. From 2003 to 2004, he was a Visiting Scholar at the School of Electrical Engineering and Computer Science, Oregon State University, USA. He is currently a Professor with the School of Electronic and Electrical Engineering, Kyungpook National University. His research interests include computer networks, multimedia, and information security.