IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 December 2023, accepted 18 January 2024, date of publication 24 January 2024, date of current version 8 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3357938

RESEARCH ARTICLE

Space Authentication in the Metaverse: A Blockchain-Based User-Centric Approach

JUNGWON SEO[1], HANKYEONG KO[2], AND SOOYONG PARK[1]
[1]Department of Computer Science and Engineering, Sogang University, Mapo-gu, Seoul 04107, South Korea
[2]Graduate School of Metaverse, Sogang University, Mapo-gu, Seoul 04107, South Korea

Corresponding author: Sooyong Park (sypark@sogang.ac.kr)

This work was supported in part by the 2024 Ministry of Science and Information and Communication Technology (MSIT), South Korea, under the Information Technology Research Center (ITRC) Support Program Supervised by the Institute for Information and Communications Technology Planning and Evaluation (IITP) under Grant RS-2024-00259099; in part by the 2024 MSIT under the Graduate School of Metaverse Convergence Support Program Supervised by the IITP under Grant RS-2022-00156318; and in part by the 2024 Culture, Sports and Tourism Research and Development Program through the Korea Creative Content Agency funded by the Ministry of Culture, Sports and Tourism, South Korea, in 2024, under Grant RS-2023-00219237.

ABSTRACT As the metaverse gains attraction, the importance of metaverse security research becomes increasingly evident. While there has been research on authenticating users in the metaverse, there is a notable gap in research concerning the authentication of specific spaces within the metaverse. This paper addresses this gap by proposing a novel user-centric blockchain-based authentication approach that incorporates space authentication. The proposed approach leverages blockchain smart contracts to authenticate users using cosine similarity metrics. A significant advantage of this approach its ability to establish user-centric authentication by seamlessly integrating metaverse and blockchain technologies, all without the need for a centralized authority. In this paper, we not only evaluate the security of our proposed approach but also conduct experiments to determine the cosine similarity threshold and assess its feasibility within a metaverse environment.

INDEX TERMS Blockchain, metaverse, authentication based on blockchain, user authentication, space authentication.

I. INTRODUCTION

The metaverse is a virtual realm where individuals lead their daily lives and engages in economic activities through avatars representing their real-life identities [1]. It is a multidimensional spatial world where users can interact with each other [2] and constitutes a fusion of all digital spaces interconnected through the internet [3].

The metaverse is often defined as an infinite digital realm centered around multiple users who access it from the real world [4]. Its primary hallmark is the ability to offer diverse experiences to users, free from the constraints of geographical barriers, space, and time. For instance, in the field of medicine, technologies such as medical twin [5] create virtual equivalents of real-world environments to forecast the outcomes of surgical procedures and treatments, while Mesh [6] establishes virtual offices for conducting meetings.

As interest in metaverse has surged, extensive research has been conducted on the constituent technologies comprising metaverse [7], [8], as well as on harnessing metaverse [9], [10]. Specially, considerable efforts have been devoted to the authentication of metaverse users [11], [12], [13].

Yao, Yingying's research [14] advocates the use of decentralized identifiers and trusted authority for authenticating users' real identities and linking them to the metaverse. Meanwhile, Samira Bader's research [15] focuses on authenticating metaverse users based on their biometric information.

In addition to the earlier-discussed research efforts, there is a wide array of ongoing studies in the field of user authentication for the metaverse. These investigations tend to primarily revolve around the authentication of users as they transition from physical world to the metaverse environment. They explore methods to securely verify the

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

identity of users within the metaverse system. However, many of these studies often overlook the intricate spatial aspects of the metaverse, which could potentially offer unique opportunities for authentication across various system and user interactions. Furthermore, it is worth noting that existing research in the realm of authentication within the metaverse often falls short in considering the multifaceted scenarios that can emerge within these virtual spaces.

Consider, for instance, a metaverse healthcare system where the need arises to demarcate distinct accessible areas for healthcare workers, patients, and nurses. Similarly, within a metaverse office environment, it might be imperative to delineate access based on specific job roles. Notably, even platforms like Decentraland's Meta Miner [16] have orchestrated promotions within the metaverse, effectively segregating spaces to grant exclusive access solely to certain individuals, such as VIP.

In the context of segregating spaces within a metaverse environment, there are two prevalent approaches: assigning a specific rank to a user's avatar [17] and implementing password protection for a space [18]. The rank assignment method involves the metaverse system determining the avatar's rank at the time of user character creation, with access to specific spaces contingent upon the user's authorized rank. Conversely, the password method entails furnishing a user with a password, granting access to a designated space exclusively to those who possess knowledge of the password. While both of these technique facilitate user authentication for space access, they are not without their vulnerabilities. There exist a risk that a malicious user may tamper with avatar's assigned rating [19] or divulge the space password [20].

Ongoing authentication research is exploring novel approaches that deviate from conventional methods. Within this realm, two predominant categories of research have emerged: 1) user role-based space authentication [21], [22], [23], [24], and 2) policy-based space authentication [25], [26], [27]. User role-based space authentication involves the assignment of specific roles to users seeking authentication for access to a space, contingent upon their designated role. Conversely, space policy authentication researchers ascribe distinct policies to spaces requiring authentication, with user authentication being governed by the applicable policy when accessing the space. A noteworthy limitation prevalent across these existing studies lies in the centralization of role assignment for users or policy establishment for spaces.

This paper presents a novel user-centric authentication technique based on blockchain technology, addressing the limitation inherent in existing research approaches. Traditional techniques often come with vulnerabilities, including the potential for credential compromise or alteration, as well as the risk of user information exposure to centralized authorities. This user-centric approach prioritizes the security and autonomy of individual users in the authentication process, ensuring that their credentials are managed in a way that minimizes exposure to external threats and vulnerabilities. Additionally, it empowers users by giving them direct control over their authentication data, thereby reducing reliance on centralized systems that can be points of failure or targets for malicious attacks.

Our proposed approach introduces a mechanism wherein service users autonomously generate their own authentication tokens using blockchain technology. These tokens are subsequently authenticated through a blockchain smart contract to secure access to specific spaces. In this approach, service users take charge of creating and managing their authentication credentials, mitigating external theft risks. Furthermore, the service provider maintains the flexibility to modify space settings as needed through the smart contract.

The contribution of this paper is as follows:

- Presenting a user-centric authentication approach leveraging blockchain technology.
- Introducing an authentication method tailored to the space dimension of the metaverse.
- Suggesting a practical convergence strategy between the metaverse and blockchain.
- Through experimental validation, affirming the direct applicability of this authentication method within metaverse environment.

The remainder of this paper is organized as follows. Section II presents the background knowledge, and Section III reviews the related work. Section IV explains the proposed approach, and Section V discusses the experiments conducted to demonstrate the efficacy of the proposed approach. Section VI concludes the paper and discusses future work.

II. PRELIMINARIES

A. THREE-DIMENSIONAL COORDINATE SPACE

Metaverse can be organized in either two-dimensional (2D) or three-dimensional (3D) configurations. In the case of 3D metaverses, spatial positioning is defined by three axes, constituting the third dimension perceptible by humans. The 3D space is comprised of the x, y, and z axes, with each point, line, and plane in this space being uniquely described by these three axes. The point where all three axes intersect is commonly referred to as the origin, and any coordinate within three-dimensional space can be accurately defined with respect to the x, y, and z axes originating from this central point.

B. COSINE SIMILARITY AND HYPERBOLIC TANGENT

The angle between two vectors in a multidimensional space can be expressed using cosine values, and it is the cosine similarity that serves as an indicator of the similarity between two specific vectors [28], [29]. Cosine similarity yields values in the range of 0 to 1, where a similarity of 1 signifies complete identity between the two vectors, while a similarity of 0 indicates no similarity.

$$\text{cosine\_similarity}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \tag{1}$$

Equation 1 provides the formula for computing cosine similarity. In this equation, A and B represent vectors, and the dot (·) signifies the dot product of these vectors. The variables A and B denote the magnitudes (norms) of vectors A and B, respectively.

The hyperbolic tangent function, often denoted as *tanh*, is utilized to map real numbers onto a bounded range spanning from -1 to 1 [30]. Notably, as values of x increase in magnitude around the origin, they asymptotically approach 1, whereas smaller x values converge to -1. This function is occasionally classified as a type of sigmoid function due to its characteristic S-shaped curve.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2}$$

Equation 2 presents the formula for the hyperbolic tangent function. In this equation, x represents the input variable passed to the hyperbolic tangent function, with the function utilizing the value of x during calculation. e^x represents the exponential function of x, where e^x denotes the base of the natural logarithm raised to the power of x. Additionally, e^{-x} signifies the negative exponential function of x, with e^x raised to the negative value of x.

C. BLOCKCHAIN

Blockchain is a decentralized technology designed to address the limitations of traditional centralized systems by ensuring the integrity and transparency of stored data. Issues such as the risk of a single point of failure and the potential for data leakage, which are associated with centralized systems, can be effectively mitigated through the utilization of blockchain.

Blockchain can be broadly classified into three categories: public, consortium, and private blockchain [31]. A public blockchain, also known as permissionless blockchain, permits unrestricted participation from anyone in the network. While public blokchains are characterized by high levels of decentralization, they exhibit certain drawbacks in terms of privacy, security, and performance. Prominent examples of public blokchain include Bitcoin and Ethereum.

A consortium blockchain is classified as permissioned blockchain. In this arrangement, each organization within the consortium functions as a node on the blockchain, and any external organizations seeking to participate in the blockchain must obtain permission from the consortium. Consortium blockchains, while offering less decentralization compared to public blockchains, excel in terms of performance, as exemplified by platforms like Hyperledger Fabric.

Private blockchains, on the other hand, operate as closed, invitation-only networks composed of pre-approved participants. These networks offer enhanced security and superior performance compared to consortium blockchains. Despite their closed nature, private blockchains permit users to access data within the blockchain network, with data generation facilitated thorough specific endpoint participants.

D. SMART CONTRACT

First introduced in the 1990s by Szabo, the concept of a smart contract represents a computerized processing protocol designed to enforce contractual terms [32]. Ethereum pioneered the implementation of smart contracts, which operate based on predefined code and automatically execute when specified conditions are met. This innovation reduces on trusted intermediaries and mitigates the risk of transaction fraud [33].

Given that a smart contract is essentially code designed to operate automatically upon meeting specific conditions within the blockchain, it offers distinct functionalities that can only be accessed through specific keywords.

The *Constructor* keyword represents a function that can be optionally utilized during the creation of a smart contract. This function is executed only once upon the contract's creation and deployment. In addition, the *Require* keyword serves as a function that verifies a particular condition within a function and forcibly terminates the smart contract if the condition is not satisfied. Additionally, the *Modifier* keyword is employed for applying supplementary rules to a smart contract function. By employing *Modifier*, users can assess preconditions or post process actions while safeguarding the function's core content.

III. RELATED WORK

In this section, we delve existing research on authentication methods for accessing spaces. Existing research can be broadly categorized into two main groups: 1) user role-based space authentication [21], [22], [23], [24] and policy-based space authentication [26], [27].

Zhu et al. and their colleagues conducted a study [21] in which they utilized computer vision algorithms to create a 3D model of physical space, aiming to enhance space authentication. Their approach involved the mapping of 3D space coordinates to the physical real-world space for user location identification. Additionally, in this approach, roles determining user authorization for specific locations were assigned by a central entity. Subsequently, users were granted access to specific spaces based on their pre-defined roles.

Another study by Wright and Madey [22] emphasized the necessity of restricting access to spaces and objects, particularly in virtual reality games and educational settings. They proposed a method that allowed movement within a specific space and interaction with objects through discretionary access control. In this research, users were categorized into group roles or individual user roles, and their access to specific spaces was determined by their assigned roles.

Wei et al. and their collaborators introduced a study [23] that focused on assigning identity-based capabilities to users, enabling or restricting their access to specific spaces based on these capabilities. This approach utilized identification-based access control technology for access control and user authentication.

Sun et al. and their team presented a study [24] addressing the lack of appropriate and flexible access control and

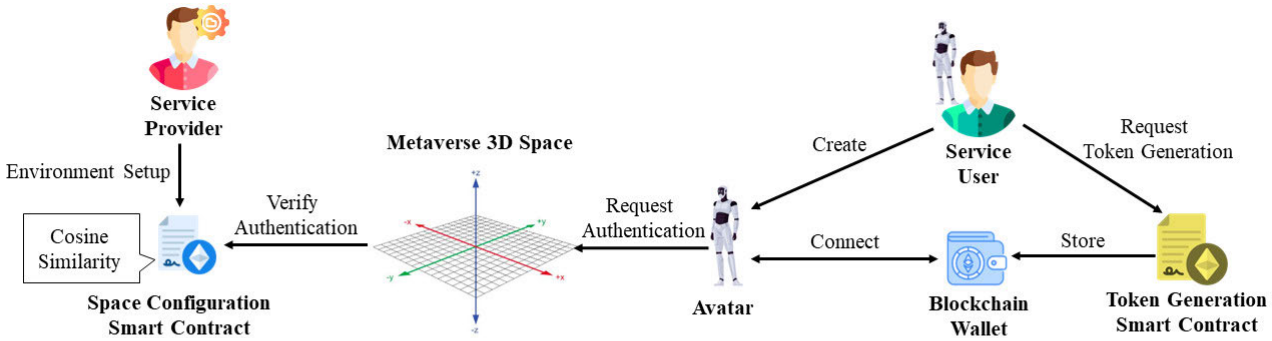


FIGURE 1. Approach overview.

authentication mechanisms in 3D virtual environments. They proposed a method for authenticating spaces using role-based access control technology. In this research, role-based access control technology was employed to enable users to authenticate themselves for accessing objects, assets, and avatars within the space.

It's important to note that the aforementioned studies [21], [22], [23], [24] all involved assigning specific roles to users for authentication when accessing particular spaces. However, a common limitation in these studies is the centralization of role assignment, which may lead to vulnerabilities such as a single point of failure and potential theft of user-assigned roles and information by malicious actors.

Lehaman and Tan [25] introduced a study that suggests authenticating users by enforcing specific rules within a given space. This research involved determining the user's physical location using GPS data from their cell phone within the real-world environment. By combining various location-based information, user authentication was attempted when accessing a particular space, relying on predefined policies associated with that space.

Tsankov et al. and their team proposed a study [26] that divided the physical environment into distinct spaces and established global requirements for each of these spaces. The research introduced a synthesizer that combined the global requirements with individual space characteristics, subsequently applying policies to each local space. User authentication was then carried out based on the policies assigned to each individual space.

In another study, Adrian Bullock, Steve Benford, and their colleagues [27] explored user authentication through the creation of an access graph defined by space boundaries. This research involved schematizing the virtual environment according to the space boundaries and utilizing this access graph to assign policies to various spaces, determining which users could access each space.

The common theme among these studies [25], [26], [27] is the application of policies to spaces for user authentication. They sought to categorize spaces and establish fine-grained policies for each space. However, these studies, like those that assign role to users [21], [22], [23], [24], encountered the limitation that space attributes were determined by a centralized entity. Consequently, this approach allowed for potential unauthorized changes to space attributes by specific entities in an opaque manner, making it challenging to detect changes to space properties caused by malicious hackers.

The approach presented in this paper aims to address the limitations identified in existing studies, which include the risk of user information theft and leakage due to a single point of failure and the potential for space configuration to be altered clandestinely by specific central entities. The proposed solution leverages blockchain-based spatial authentication techniques to enhance security and transparency in a metaverse environment. The Table in 1 provides a comparison of related works and our proposed approach.

TABLE 1. Comparison of related works.

Reference Number	Decentralization	Consider space Attributes
[21]	X	X
[22]	X	X
[23]	X	X
[24]	X	X
[25]	X	O
[26]	O	X
[27]	X	O
Our Approach	O	O

IV. APPROACH

In this paper, we introduce a blockchain-based user-centric authentication approach. Our proposed method empowers users to generate their authentication credentials, granting them access to specific metaverse spaces through a blockchain smart contract. The authorization process we propose involves two primary participants: the service provider (*SP*), responsible for offering services, and the service user (*SU*), who utilizes metaverse services. The *SP* assumes the role of an individual accountable for designing and managing a metaverse system.

The term *metaverse system* denotes a specific metaverse environment that is conceived and managed by an *SP*. Conversely, *SU* represents an individual who utilizes the

metaverse system offered by the *SP*. The *SU* is pre-authorized to access particular metaverse spaces through their avatar, a privilege facilitated by the *SP*. Furthermore, the proposed approach operates under the assumption of utilizing a consortium or private blockchain, where blockchain data is not readily accessible, as opposed to a public blockchain.

Also, in our proposed approach, alongside the two primary participants, *SU* and *SP*, two distinct smart contracts are deployed on the blockchain. The *Space Configuration Smart Contract* serves as a specialized smart contract utilized by the *SP* for space authentication. Furthermore, the *Token Generation Smart Contract* is dedicated to the generation and administration of authentication tokens initiated by the *SU*.

Figure 1 provides a simplified illustration of the process within our proposed approach. The *SP* deploys a *Space Configuration Smart Contract* on the blockchain while constructing a metaverse system. This *Space Configuration Smart Contract* is responsible for verifying user authentication when an *SU* presents an authentication token through cosine similarity. The *SU*, on the other hand, deploys a *Token Generation Smart Contract* to generate an authentication token and store it in the *SU*'s blockchain wallet. When the *SU*'s avatar accesses a space requiring authentication, the avatar submits the *SU*'s authentication token from the blockchain wallet to the metaverse system. Subsequently, the metaverse system forwards the token information to the *Space Configuration Smart Contract* for verification.

TABLE 2. Notation in this paper.

Symbol	Symbol Definition
SP	Service Provider
SU	Service User
$Sreq_{auth}$	Space requiring authentication
P	Set of properties
Ps_n	Elements in P. n is the order of the elements
C	Condition for accessing the space
Tan_c	C with hyperbolic tangent
$User_c$	User-created conditions
$TanUser_c$	$User_C$ with hyperbolic tangent

The proposed approach can be categorized into three primary components: 1) *SP*'s methodology for metaverse space authentication design, 2) *SU*'s authentication token generation process, and 3) Authentication of *SU* avatar. Each of these components is further detailed in the respective subsections. Additionally, corresponding notation for each process is provided in Table 2.

A. DESIGNING METAVERSE SPACE AUTHENTICATION FOR SERVICE PROVIDER

This section outlines the procedure through which service provider (*SP*) establish authentication-required spaces within the metaverse and define the authentication criteria for these spaces. Figure 2 provides a simplified illustration of this process.

The *SP* identifies specific spaces within the metaverse environment to which additional authorization is intended

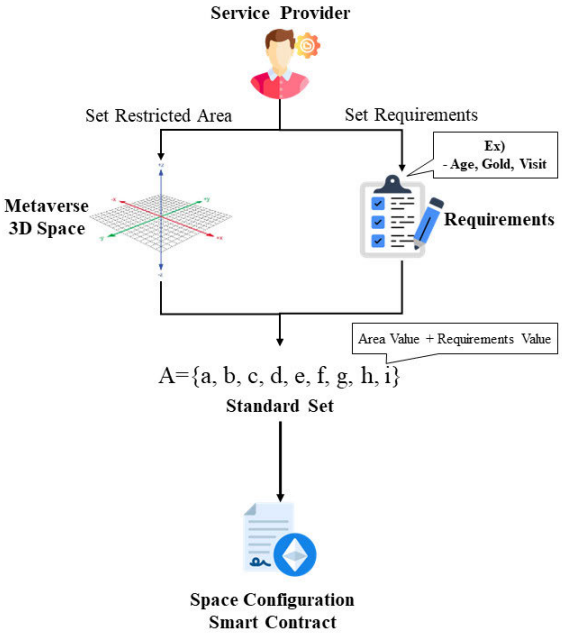


FIGURE 2. Space authentication design overview.

to be granted. This can be represented as $Sreq_{auth} = \{X$ Coordinate $\times$ Y Coordinate $\times$ Z Coordinate$\}$. Subsequently, the *SP* defines the attributes eligible for accessing $Sreq_{auth}$, creating a set denoted as $P = \{Ps_1, Ps_2, Ps_3 \cdots Ps_n\}$.

Following this, the *SP* shares the P set, containing authorized attributes, with *SU*, granting *SU* permission to access $Sreq_{auth}$. The *SP* then employs the defined $Sreq_{auth}$ and attribute set P to create authorization context, denoted as C. This authorization context encompasses critical details, including X and Y minimum and maximum coordinates, as well as Z minimum and maximum coordinates. Additionally, it includes the attributes Ps_1 through Ps_n.

In the process of selecting attributes (Ps), the *SP* must exercise diligence in choosing values that can be seamlessly verified within the metaverse system. For instance, if the *SP* constructs a metaverse within the space range of $X = [0, 100]$, $Y = [0, 100]$, $Z = [0, 100]$, the specific authorization scope they intend to establish can be defined as follows: $X = [0, 20]$, $Y = [0, 25]$, $Z = [0, 30]$, resulting in $Sreq_{auth} = [0, 20] \times [0, 25] \times [0, 30]$. Subsequently, the *SP* proceeds to specify the attributes that grant access to $Sreq_{auth}$, forming a set denoted as $P = \{$User age > 35, Avatar Gold > 3000, access log $> 100\}$, which is then share with the *SU*. Following this, the *SP* generates an authorization context C, encompassing crucial details such as X minimum and maximum coordinates, Y minimum and maximum coordinates, Z minimum and maximum coordinates, user age > 35, avatar gold > 3000, and access log > 100, resulting in $C = \{0, 20, 0, 25, 0, 30, 35, 3000, 100\}$.

Afterwards, the *SP* substitutes the generated authorization context C into the hyperbolic tangent formula, replacing the numerical values. Applying the aforementioned example to

the hyperbolic tangent function, C is transformed into Tan_c = {0.9999999958777, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0}. The hyperbolic tangent function is a mathematical operation that maps any real number to a value within the range of −1 to 1. Utilizing the hyperbolic tangent function allows for the establishment of a bounded value range, simplifying the computation of cosine similarity. Furthermore, the hyperbolic tangent function can serve as an initial layer of security for the authentication attribute values, as it makes it challenging for malicious users to reverse-engineer input values from output values, thus deterring them from easily discerning the authentication attributes.

The *SP* stores the computed Tan_c values within a blockchain smart contract known as the *Space Configuration Smart Contract* (*SCSC*). The *SCSC* safeguards against unauthorized access by users other than the *SP* through the utilization of *Constructor*, *Require*, and *Modifier* functions. The procedure for storing Tan_c within the *SCSC* is detailed in Algorithm 1.

Algorithm 1 Designing the Space Configuration Smart Contract for Storing Tan_c

Require: Tan_c
  Constructor (owner == msg.sender)
  Modifier (msg.sender == owner)
  if modifier check == true **then**
    Data save (Tan_c)
    Data call (Tan_c)
  end if
  if modifier check == false **then**
    Revert
  end if

When deploying the *SCSC*, the contract owner is assigned to the *SP* since the *SP* is responsible for deploying the *SCSC*. Once the *SCSC* is deployed, it can receive requests for access from various users. A verification process is then conducted to confirm whether the blockchain address of the requester matches the one stored within the smart contract for the *SP*. If the request originates from the same blockchain address as the *SP*, the smart contract grants permission for storing, modifying the Tan_c. However, if the request comes from a different blockchain address, all access requests are denied.

B. AUTHENTICATION TOKEN GENERATION FOR SERVICE USER

This section explains the procedure for a service user (*SU*) to generate an authentication token for accessing authenticated spaces, as illustrated in Figure 3.

Since there is no centralized authority in the proposed approach, *SU*s are tasked with issuing and managing their own tokens. The *SU* generates a token based on the predefined conditions within P, which have been communicated by the *SP* in advance.

For instance, if the *SU* is aware of $Sreq_{auth}$ = {[0, 20] × [0, 25] × [0, 30]} and understands that the prerequisite

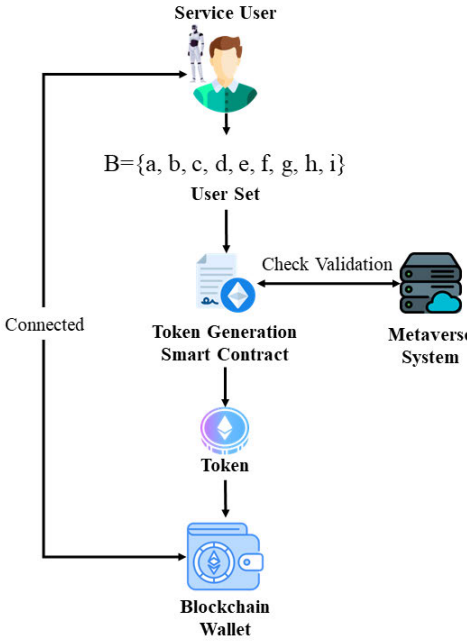


FIGURE 3. Authentication token generation overview.

conditions are P = {User age > 35, Avatar Gold > 3000, access log > 100} for accessing $Sreq_{auth}$, the *SU* can generate $User_c$ = {0, 20, 0, 25, 0, 30, 40, 5000, 200}.

The generation of authentication tokens is executed through the *Token Generation Smart Contract* (*TGSC*), with the process being outlined in Algorithm 2.

Algorithm 2 Designing Token Generation Smart Contract

Require: $User_c$
  Constructor (owner == msg.sender)
  Modifier (msg.sender == owner)
  Disassemble ($User_c$)
  Verify X, Y, Z coordinates are in the Metaverse
  if Verify == true **then**
    Verify $Ps_1, Ps_2, Ps_3, \ldots, Ps_n$
    if Ps are valid **then**
      Generate $TanUser_c$ ($User_c$)
      Generate authentication token ($TanUser_c$)
      Store to user's Wallet
    end if
  end if

When the *SU* transmits the C value, the *TGSC* initiates communication with the metaverse system to validate the correctness of the $User_c$ value. It conducts check to ensure that the specified space coordinate values provided by the *SU* correspond to valid spaces within the metaverse. Additionally, it verifies whether the P values supplied by the *SU* align with the *SU*'s authenticated information within the metaverse system.

If the metaverse system determines that all $User_c$ values presented by the *SU* are valid, the *TGSC* applies the

hyperbolic tangent function to the $User_c$ value to create $TanUser_c$. Subsequently, the *TGSC* generates an authorization token utilizing $TanUser_c$ as metadata and stores it in the *SU*'s wallet. The *TGSC* incorporates the same *Modifier* and *Constructor* functions as the *SCSC* to ensure that only the correct *SU* can access the *TGSC* and read the token's metadata values.

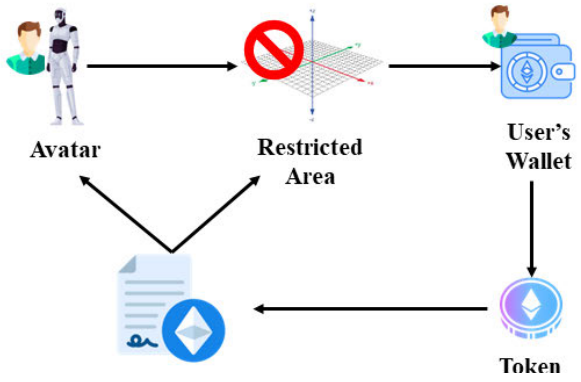


FIGURE 4. Authentication overview.

C. AUTHENTICATION OF SERVICE USER AVATARS

This section outlines how a *SU* utilizes avatar in the metaverse to access and authenticate to authenticated spaces, as demonstrated in Figure 4. When the *SU*'s avatar attempts to access the *SP*'s established parameters for $Sreq_{auth}$, the metaverse system activates the avatar's blockchain wallet. The user then send the stored authorization token from the wallet to the metaverse system through the avatar. The metaverse system further forwards this authentication token to the *SCSC* for a verification process to assess its suitability for authentication.

The *SCSC* initially verifies the validity of the requested address through the *TGSC*. If the avatar's blockchain address is confirmed as valid by the *Constructor* and *Modifier* functions, the *TGSC* grants permission to the *SCSC* to access $TanUser_c$. Subsequently, the *SCSC* vectorizes both Tan_c and $TanUser_c$, facilitating the calculation of cosine similarity between the two vectors. The cosine similarity metric offers a measurement of how closely $TanUser_c$ aligns with the reference Tan_c stored within the *SCSC*. The resulting similarity score ranges from 0 to 1, with a score of 1 indicating an exact match with the reference Tan_c and scores closer to 1 indicating a higher degree of similarity. This similarity value forms the basis for determining whether the service user qualifies for access to the authenticated space.

The *SCSC* computes the cosine similarity between the Tan_c and $TanUser_c$ values. If the similarity result surpasses the defined similarity tolerance threshold (*Sa*), the *SCSC* concludes that $TanUser_c$ possesses attribute values sufficiently akin to Tan_c. Subsequently, the *SCSC* notifies the metaverse system to proceed with user authentication, and the metaverse system grants authorization to the avatar, allowing access to the designated space.

V. EXPERIMENTS

In this section, we formulate four research questions pertaining to our approach and present the outcomes of four experiments aimed at addressing these questions:

- *RQ1)* How can the security of the proposed approach be ensured?
- *RQ2)* What are the False Acceptance Rate (FAR) and False Rejection Rate (FRR) in relation to varying similarity thresholds?
- *RQ3)* What is the error rate associated with the similarity threshold identified in *RQ2*?
- *RQ4)* What is the authentication time within an authentic metaverse environment employing the proposed approach?

A. RQ1) HOW CAN THE SECURITY OF THE PROPOSED APPROACH BE ENSURED?

In this section, we present a comprehensive analysis aimed at assessing the security of the proposed approach. In our security evaluation, we elucidate the threat model and elaborate on how the proposed approach effectively mitigates impersonation attacks, safeguards confidentiality, and ensures non-repudiation.

1) THREAT MODEL

In the proposed approach, users create blockchain-based authentication tokens for accessing designated spaces. Within this context, and adversary ($\mathbb{A}$) may attempts an impersonation attack, aiming to deceive the system into perceiving them as a legitimate user. Additionally, $\mathbb{A}$ may engage in the tampering of Tan_c, which forms the foundation of authentication, thereby hindering the proper authentication of genuine users termed a confidentiality breach attack. Furthermore, even after authentication, $\mathbb{A}$ might still initiate a non-repudiation attack.

2) IMPERSONATION ATTACK

An impersonation attack entails $\mathbb{A}$ assuming the identity of another user and infiltrating the system by mimicking normal user behavior. To execute such an attack, $\mathbb{A}$ must possess prior knowledge of the *Ps* values shared between *SP* and *SU*. In essence, if the *SP* establishes *Ps* values that are sufficiently complex, $\mathbb{A}$ would encounter significant difficulties attempting to access the system incognito as a regular user.

Another potential avenue for $\mathbb{A}$ to achieve success in an impersonation attack involves the theft of a legitimate user's blockchain private key and authentication token. Blockchain private keys are typically generated using 12 to 24 words, following the mnemonic code of BIP-0039 [34], comprising 2048 characters. In practical terms, this implies that $\mathbb{A}$ would need to correctly match 12 to 24 words to pilfer a specific user's private key and execute a successful impersonation

attack. Given the vast number of possible combinations - approximately 5.27×10^{39} ($\approx 2048!\ /\ (2048-12)!$) - for generating a private key with 12 words, it becomes evident that $\mathbb{A}$'s chances of succeeding in an impersonation attack are exceedingly slim.

Furthermore, $\mathbb{A}$ may attempt to access Tan_c or $TanUser_c$ store in the blockchain data for impersonation attacks. As outlined in the approach, to access Tan_c or $TanUser_c$, a correct address must be provided to the smart contract. Acquiring a proper address necessitates the possession of a corresponding private key. However, as discussed in the preceding paragraph, it is challenging for $\mathbb{A}$ to obtain a valid private key through random or luck-based methods.

3) CONFIDENTIALITY

In our proposed approach, the potential for confidentiality breach arises if $\mathbb{A}$ attempts to alter Tan_c, the foundation of the authentication process, thereby obstructing the proper authentication of legitimate users. To compromise confidentiality, $\mathbb{A}$ would need to gain access to the *SCSC* and modify Tan_c. However, this modification requires authorization through a smart contract. Within the *SCSC* design articulated in our approach, data access is exclusively granted to the *SP* responsible for the initial distribution of the *SCSC*.

Essentially, this implies that $\mathbb{A}$ cannot breach confidentiality since data security is assured by the blockchain smart contract. Furthermore, even if $\mathbb{A}$ manages to manipulate the data within the *SCSC*, the transparency and immutability inherent to blockchain technology enable the *SP* to promptly detect alterations and trace $\mathbb{A}$'s activities.

4) NON-REPUDIATION

Authentication in the proposed approach is facilitated via the blockchain network. Every transaction and record within the blockchain is transparently logged, and their integrity is safeguarded through cryptographic hash functions. When a user undergoes authentication, their authentication record is immediately stored within the blockchain. Therefore, even if $\mathbb{A}$ were to authenticate and subsequently attempt non-repudiation, $\mathbb{A}$ would be unable to engage in non-repudiation because all records pertaining to $\mathbb{A}$ are securely recorded in the blockchain and accessible to each network node.

B. RQ2) WHAT ARE THE FALSE ACCEPTANCE RATE (FAR) AND FALSE REJECTION RATE (FRR) IN RELATION TO VARYING SIMILARITY THRESHOLDS?

In this section, an experiment was conducted to determine the appropriate similarity tolerance *Sa* setting when measuring the cosine similarity between $TanUser_c$ and Tan_c. To carry out this experiment, we established a standard Tan_c with three attribute values, denoted as $P = \{Ps_1, Ps_2, Ps_3\}$. Additionally, we generated 50,000 sets of $TanUser_c$s and 50,000 sets of unauthorized $TanUser_c$s based on the standard Tan_c. The objective was to assess the impact of varying *Sa* values on the False Acceptance Rate (FAR) and False Rejection Rate (FRR).

Preparation involved generating 100 different *Sa* values for experiment, ranging from 0 to 1 in increments of 0.01. For each generated *Sa*, the cosine similarity between the $TanUser_c$, which should be accepted according to the data set, and the standard Tan_c was computed. If the similarity value exceeded the *Sa* threshold, it was categorized as an acceptance error, and the error ratio out of 50,000 cases was computed to determine the FAR. Conversely, if the similarity value fell below the *Sa* threshold, it was regarded as a rejection error, and the error ratio out of 50,000 cases was calculated to establish the FRR.

The results of the FAR measurements are depicted in Figure 5. These results indicate that as the *Sa* increases, the FAR decreases. When the *Sa* value exceeds 0.7, the FAR approaches zero, signifying a lower likelihood of accepting unauthorized access attempts.

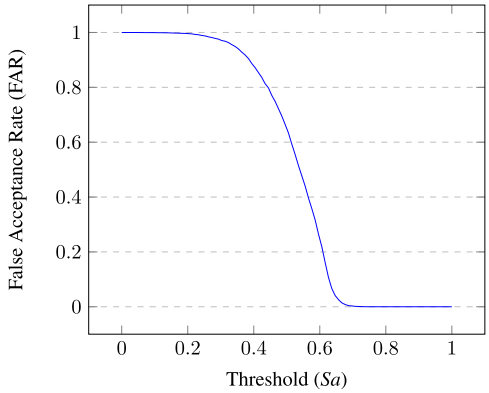


FIGURE 5. FAR based on Threshold.

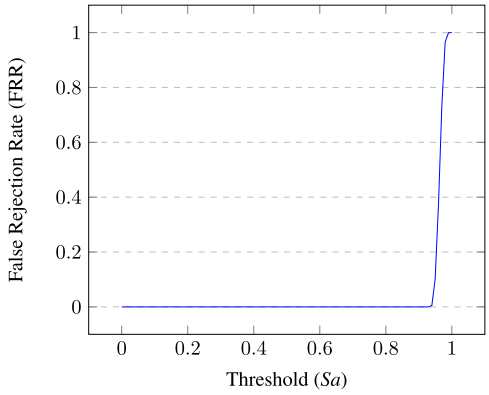
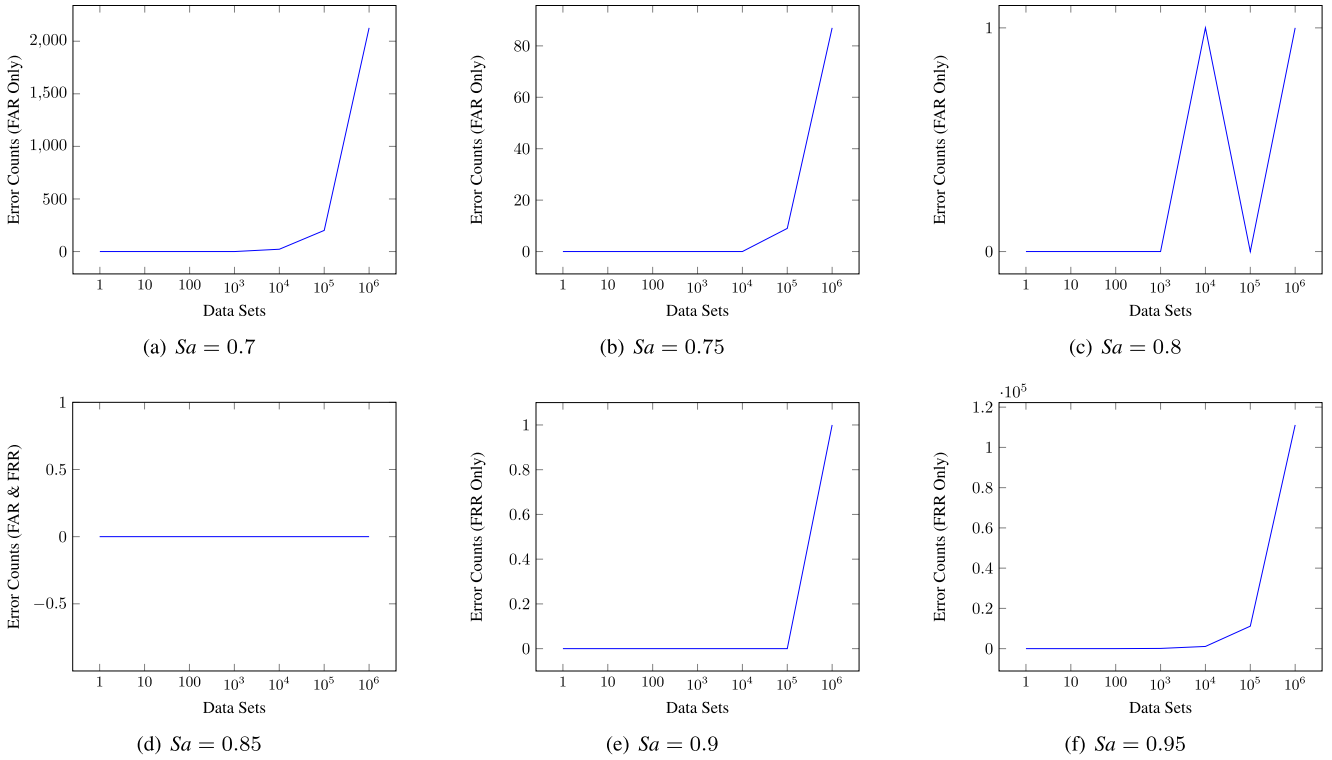


FIGURE 6. FRR based on Threshold.

The FRR results are displayed in Figure 6. These FRR measurement results reveal that as the *Sa* increases, the FRR also rises. Notably, the FRR value experiences a significant spike when the *Sa* value reaches 0.9.

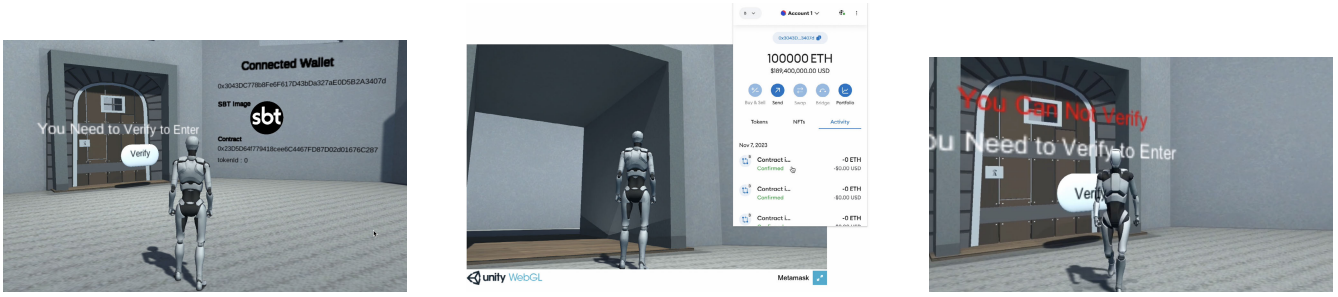
The FAR and FRR measurements demonstrated that the lowest FAR and FRR values were achieved within the *Sa* range of 0.7 to 0.95.



(a) $Sa = 0.7$ (b) $Sa = 0.75$ (c) $Sa = 0.8$

(d) $Sa = 0.85$ (e) $Sa = 0.9$ (f) $Sa = 0.95$

FIGURE 7. Error Counts based on *Sa*.



(a) Authentication Request (b) Authentication Success (c) Authentication Failure

FIGURE 8. Metaverse implementation.

C. RQ3) WHAT IS THE ERROR RATE ASSOCIATED WITH THE SIMILARITY THRESHOLD IDENTIFIED IN RQ2?

RQ2 indicated that the lowest FAR and FRR values were achieved within the *Sa* range of 0.7 to 0.95. Consequently, in this section, we conducted an experiment to assess the actual error counts within this *Sa* range. In this experiment, a standard Tan_c was created, maintaining the number of condition *Ps* at three, as in *RQ2*. We examined the occurrence of errors as the number of $TanUser_c$ that should be authorized increased from 1 to 1 million.

In this experiment, we observed that only FAR occurred when *Sa* ranged between 0.7 and 0.8, while solely FRR occurred after *Sa* reached 0.9. The experimental results, depicted in Figure 7, illustrate the number of errors on the y-axis against the number of data sets on the x-axis.

In particular, when *Sa* was set to 0.7, a total of 2,300 errors were recorded, while at an *Sa* of 0.75, there were 96 errors. The *Sa* value of 0.8 resulted in only 2 errors, 1 error for 0.9, and a significantly higher 122,197 errors for 0.95. Interestingly, no errors were observed when *Sa* was set to 0.85, indicating that the proposed approach achieves its highest authentication accuracy at an *Sa* value of 0.85.

D. RQ4) WHAT IS THE AUTHENTICATION TIME WITHIN AN AUTHENTIC METAVERSE ENVIRONMENT EMPLOYING THE PROPOSED APPROACH?

In this section, our objective was to evaluate the feasibility of implementing the proposed approach within a metaverse environment. To accomplish this, we constructed a metaverse environment and conducted experiments to investigate how

the number of *Ps* (authorization attributes) influences the authentication time. The measurement of authentication time involved tracking the duration from the initiation of an authentication request to its approval.

The experiments were conducted on a computer running macOS Monterey, equipped with an Apple M1 Pro processor, 16GB of memory, and an Apple M1 Pro graphics unit. The development tools and technologies used to build the metaverse environment included the Unity 3D Engine for creating the overall environment, the Starter Assets - Third Person Character Controller Package for implementing avatar movement in 3D space, Unity C# Script, and ChainSafe SDK for integrating the blockchain network, as well as Hyperledger Besu for establishing a private blockchain network connection with Metamask, a blockchain wallet.

To access the created metaverse, users initially log in via Metamask integration and are transported to the 3D environment. Their wallet and token information are seamlessly transmitted to our system. In the virtual room, access to the authentication-restricted space is blocked by a closed door. By interacting with the 'Verify' button using their avatar, the *SCSC* required for authentication is invoked through the linked wallet, initiating the authentication process. After successful authentication, the door disappears, granting the avatar entry into the restricted spaces.

Figure 8 illustrates an instance of the metaverse environment. In Figure 8(a), there is an authentication request screen for space access, Figure 8(b) displays the door opening after successful authentication, and 8(c) shows the screen that appears when authentication fails.

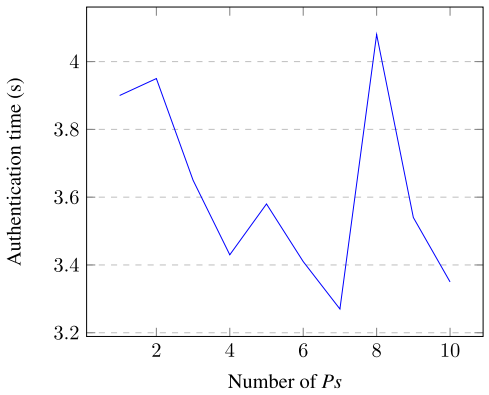


FIGURE 9. **Authentication time based on number of *Ps*.**

Figure 9 depicts the fluctuation in authentication time concerning the number of condition attribute values (*Ps*). The experimental range covered 1 to 10 *Ps* conditions, and 100 authentication tests were carried out for each *Ps* number to compute the average authentication time.

We observed that the authentication time remained relatively constant at around 3 to 4 seconds, irrespective of the number of *Ps*. The average authentication time was calculated to be approximately 3.616 seconds, and this value did not show significant variation with changes in the number of *Ps*. Therefore, it can be inferred that the number of *Ps* no substantial impact on the authentication time. The primary factor influencing authentication time appears to be the performance of blockchain network, and this is consistent regardless of the number of *Ps*.

VI. CONCLUSION

In this paper, we have introduced a user-centric authentication scheme leveraging blockchain technology for securing access to specific spaces within a metaverse environment. Our proposed approach encompasses three key components: a methodology for metaverse and smart contract design by service providers, an authentication token generation method for service users, and a space authentication method reliant on authentication tokens. We have demonstrated that this approach effectively utilizes user attributes for authentication, using cosine similarity to gauge the similarity of attributes.

A significant strength of our paper lies in the introduction of user-centric authentication, which is accomplished through the integration of metaverse and blockchain technologies, all without the need for a centralized authority. In our future research endeavors, we aim to explore the application of access control techniques to individual objects within the metaverse using this innovative approach. Additionally, we aim to reduce network communication costs between the blockchain and metaverse systems during the authentication process.

ACKNOWLEDGMENT

(Jungwon Seo and Hankyeong Ko contributed equally to this work.)

REFERENCES

[1] Q. Yang, Y. Zhao, H. Huang, Z. Xiong, J. Kang, and Z. Zheng, "Fusing blockchain and AI with metaverse: A survey," *IEEE Open J. Comput. Soc.*, vol. 3, pp. 122–136, 2022.

[2] L. Young, "A study on metaverse hype for sustainable growth," *Int. J. Adv. Smart Converg.*, vol. 10, no. 3, pp. 72–80, 2021.

[3] R. Cheng, N. Wu, S. Chen, and B. Han, "Will metaverse be NextG Internet? Vision, hype, and reality," *IEEE Netw.*, vol. 36, no. 5, pp. 197–204, Sep. 2022.

[4] M. Wright, H. Ekeus, R. Coyne, J. Stewart, P. Travlou, and R. Williams, "Augmented duality: Overlapping a metaverse with the real world," in *Proc. Int. Conf. Adv. Comput. Entertainment Technol.* ACM, Dec. 2008, pp. 263–266.

[5] T. Erol, A. F. Mendi, and D. Dogan, "The digital twin revolution in healthcare," in *Proc. 4th Int. Symp. Multidisciplinary Stud. Innov. Technol. (ISMSIT)*, Oct. 2020, pp. 1–7.

[6] (2023). *Microsoft Mesh, Virtual Office*. Accessed: Dec. 20, 2023. [Online]. Available: https://learn.microsoft.com/en-us/mesh/overview

[7] M. A. I. Mozumder, M. M. Sheeraz, A. Athar, S. Aich, and H.-C. Kim, "Overview: Technology roadmap of the future trend of metaverse based on IoT, blockchain, AI technique, and medical domain metaverse activity," in *Proc. 24th Int. Conf. Adv. Commun. Technol. (ICACT)*, 2022, pp. 256–261.

[8] H. Ning, H. Wang, Y. Lin, W. Wang, S. Dhelim, F. Farha, J. Ding, and M. Daneshmand, "A survey on the metaverse: The state-of-the-art, technologies, applications, and challenges," *IEEE Internet Things J.*, vol. 10, no. 6, pp. 14671–14688, May 2023.

[9] S. Bhattacharya, S. Varshney, and S. Tripathi, "Harnessing public health with 'metaverse' technology," *Frontiers Public Health*, vol. 10, p. 4452, Dec. 2022.

[10] S. E. Bibri and S. K. Jagatheesaperumal, "Harnessing the potential of the metaverse and artificial intelligence for the Internet of City Things: Cost-effective XReality and synergistic AIoT technologies," *Smart Cities*, vol. 6, no. 5, pp. 2397–2429, Sep. 2023.

[11] J. Ryu, S. Son, J. Lee, Y. Park, and Y. Park, "Design of secure mutual authentication scheme for metaverse environments using blockchain," *IEEE Access*, vol. 10, pp. 98944–98958, 2022.

[12] K. Yang, Z. Zhang, T. Youliang, and J. Ma, "A secure authentication framework to guarantee the traceability of avatars in metaverse," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 3817–3832, 2023.

[13] G. Thakur, P. Kumar, C.-M. Chen, A. V. Vasilakos, Anchna, and S. Prajapat, "A robust privacy-preserving ECC-based three-factor authentication scheme for metaverse environment," *Comput. Commun.*, vol. 211, pp. 271–285, Nov. 2023.

[14] Y. Yao, X. Chang, L. Li, J. Liu, J. Mißié, and V. B. Misié, "DIDs-assisted secure cross-metaverse authentication scheme for MEC-enabled metaverse," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2023, pp. 6318–6323.

[15] S. Bader and N. E. B. Amara, "Design of a 3D virtual world to implement a logical access control mechanism based on fingerprints," in *Proc. IEEE/ACS 14th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Oct. 2017, pp. 1239–1246.

[16] (2023). *Decentraland Metaminer*. Accessed: Dec. 20, 2023. [Online]. Available: https://events.decentraland.org/event/?id=c812ebfa-4db6-4b70-9fa4-1ff4c2dd2734

[17] A. E. Norris, H. Weger, C. Bullinger, and A. Bowers, "Quantifying engagement: Measuring player involvement in human–avatar interactions," *Comput. Hum. Behav.*, vol. 34, pp. 1–11, May 2014.

[18] M. I. Hossain and R. Hasan, "Threat model-based security analysis and mitigation strategies for a trustworthy metaverse," in *Proc. IEEE Int. Conf. Metaverse Comput., Netw. Appl. (MetaCom)*, Jun. 2023, pp. 33–40.

[19] Aiswarya, J. Raveendran, and V. Banahatti, "Behavioral attributes in password reuse: Analysis of password practices in work and personal spaces," in *Proc. 13th Indian Conf. Human-Computer Interact.* ACM, Nov. 2022, pp. 1–19.

[20] Y. Xiao, C.-C. Li, M. Lei, and S. V. Vrbsky, "Differentiated virtual passwords, secret little functions, and codebooks for protecting users from password theft," *IEEE Syst. J.*, vol. 8, no. 2, pp. 406–416, Jun. 2014.

[21] S. Zhu, H. J. Kim, M. Monge, G. E. Suh, A. Alaghi, B. Reagen, and V. Lee, "Verifiable access control for augmented reality localization and mapping," 2022, *arXiv:2203.13308*.

[22] T. Wright and G. Madey, "Discretionary access controls for a collaborative virtual environment," *Int. J. Virtual Reality*, vol. 9, no. 1, pp. 61–71, Jan. 2010.

[23] Y.-g. Wei, Y. Lu, X.-y. Hu, and B. Sun, "Research and application of access control technique in 3D virtual reality system OpenSim," in *Proc. 6th Int. Symp. Comput. Intell. Design*, vol. 2, Oct. 2013, pp. 65–68.

[24] B. Sun, J.-y. Xu, X.-m. Zhu, H.-q. Zhao, and J. Liu, "Research and application of access control technique in distributed 3D virtual environment," in *Proc. 7th Int. Conf. Comput. Sci. Educ. (ICCSE)*, Jul. 2012, pp. 1638–1643.

[25] S. M. Lehman and C. C. Tan, "PrivacyManager: An access control framework for mobile augmented reality applications," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Oct. 2017, pp. 1–9.

[26] P. Tsankov, M. T. Dashti, and D. Basin, "Access control synthesis for physical spaces," in *Proc. IEEE 29th Comput. Secur. Found. Symp. (CSF)*, Jun. 2016, pp. 443–457.

[27] A. Bullock and S. Benford, "Access control in virtual environments," in *Proc. ACM Symp. Virtual reality Softw. Technol.*, Sep. 1997, pp. 29–35.

[28] M. Sugang, L. Ningbo, P. Guansheng, C. Yanping, W. Ying, and H. Zhiqiang, "Object detection algorithm based on cosine similarity IoU," in *Proc. Int. Conf. Netw. Netw. Appl. (NaNA)*, Dec. 2022, pp. 1–6.

[29] (2023). *Cosine Similarity*. Accessed: Dec. 20, 2023. [Online]. Available: https://leonlok.co.uk/blog/finding-similar-names-using-cosine-similarity/

[30] T. K. R. Arvind, M. Brand, C. Heidorn, S. Boppu, F. Hannig, and J. Teich, "Hardware implementation of hyperbolic tangent activation function for floating point formats," in *Proc. 24th Int. Symp. VLSI Design Test (VDAT)*, Jul. 2020, pp. 1–6.

[31] T. Zhang and Z. Huang, "Blockchain and central bank digital currency," *ICT Exp.*, vol. 8, no. 2, pp. 264–270, 2022.

[32] N. Szabo. (2023). *Smart Contracts. Phonetic Sciences*. Accessed: Dec. 20, 2023. [Online]. Available: https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html

[33] Z. Zheng, S. Xie, H.-N. Dai, W. Chen, X. Chen, J. Weng, and M. Imran, "An overview on smart contracts: Challenges, advances and platforms," *Future Gener. Comput. Syst.*, vol. 105, pp. 475–491, Apr. 2020.

[34] (2023). *Bitcoin/BIPs*. Accessed: Dec. 20, 2023. [Online]. Available: https://github.com/bitcoin/bips/blob/master/bip-0039/english.txt

JUNGWON SEO received the degree major in management information system from the Business Department, State University of New York at Buffalo, in May 2016, and the master's degree in computer science and engineering from Sogang University, in March 2020, with a major in software engineering and blockchain, where he is currently pursuing the Ph.D. degree majoring in software engineering and blockchain.

HANKYEONG KO received the bachelor's degree in business administration from The Catholic University of Korea, in September 2018, and the master's degree in metaverse engineering from Sogang University, with a specialization in metaverse engineering and blockchain. He is the coauthor alongside Jungwon Seo.

SOOYONG PARK received the Ph.D. degree from George Mason University, in 1995. He has held prestigious positions, such as a Professor in computer science with Sogang University, since March 1998, and a Professor with the Metaverse Graduate School, Sogang University, since March 2022. He has been serving as the President of the Korea Society of Blockchain, since January 2019. He is also the Director of the Intelligent Blockchain Research Center since that time. Previously, he was the President and the CEO of the National IT Industry Promotion Agency (NIPA), from September 2012 to November 2014. His accolades include the Asia–Pacific Software Engineering Conference (APSEC) 10 Year-Most Influential Paper Award, in December 2018. He received the Korea Minister of Science and ICT Award for Best Project Evaluation, in November 2021.