Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4389729, 13 pages
https://doi.org/10.1155/2022/4389729

Research Article

Authenticated Wireless Links between a Drone and Sensors Using a Blockchain: Case of Smart Farming

Kahlid S. Alqarni,[1] **Faris A. Almalki**,[2] **Ben Othman Soufiene**,[3] **Obaid Ali**,[4] **and Faisal Albalwy**[5,6]

[1]*Department of Information Technology, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia*
[2]*Department of Computer Engineering, College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia*
[3]*PRINCE Laboratory Research, ISITcom, Hammam Sousse, University of Sousse, Tunisia*
[4]*Department of Computer Science & Information Technology, Ibb University, Ibb, Yemen*
[5]*Department of Computer Science, College of Computer Science and Engineering, Taibah University, Madinah, Saudi Arabia*
[6]*Division of Informatics, Imaging and Data Sciences, Stopford Building, University of Manchester, Oxford Road, Manchester M13 9PL, UK*

Correspondence should be addressed to Obaid Ali; obaid.alii2016@gmail.com

Received 23 June 2022; Revised 3 August 2022; Accepted 25 August 2022; Published 5 September 2022

Academic Editor: Yin Zhang

Agriculture is confronted with several significant difficulties, such as rising air temperatures and population growth, causing the implementation of smart farming operations as an optimum solution. This research aims to contribute to the growing knowledge of the potential role of blockchain technology in promoting the concept of smart farming by enhancing the efficiency of farming operations by boosting agricultural production, lowering environmental impact, and automating the work of farmers. It proposes a secure blockchain-based framework to establish trust among smart farming users. The framework utilizes asymmetric key exchange mechanism using an ECC authentication algorithm and SHA-256 hash function cryptography to secure communication between sensors and drones in the farm field. The SHA-256 hashing function ensures data integrity as attempts to tamper with data result in a different hash value, breaking the chain of blocks. To demonstrate the feasibility of the proposed framework, a proof-of-concept implementation was developed on the Ethereum blockchain, in which smart contracts were used to model the framework operations. The proof of concept's performance was examined using Hyperledger Caliper for latency, throughput, and transaction success rate. The findings clearly indicate that blockchain technology can provide an efficient and scalable mechanism to advance smart farming and address some of the barriers that inhibit smart farming, particularly regarding to data integrity and availability.

1. Introduction

Smart farming refers to the use of various technologies and gadgets, such as the Internet, cloud, and IoT devices. By 2050, the world's population is projected to reach 9.7 billion people, requiring greater agricultural production to feed those billion people [1]. Because of causes like as industrialization, commercial marketplaces, and residential structures being constructed on agricultural areas, the population is increasing, while agricultural land is decreasing. Using these works need to be boosted for output to feed these billions, which can be done by integrating IoT in farming as shown in Figure 1. Due to several reasons, including insect attacks, plant disease, a lack of knowledge about essential nutrients for crops, and other problems, farmers are no longer able to enjoy the benefits of their work. To eliminate these obstacles and make farming more profitable, smart, and enjoyable for farmers, technological advancement is needed [2]. In

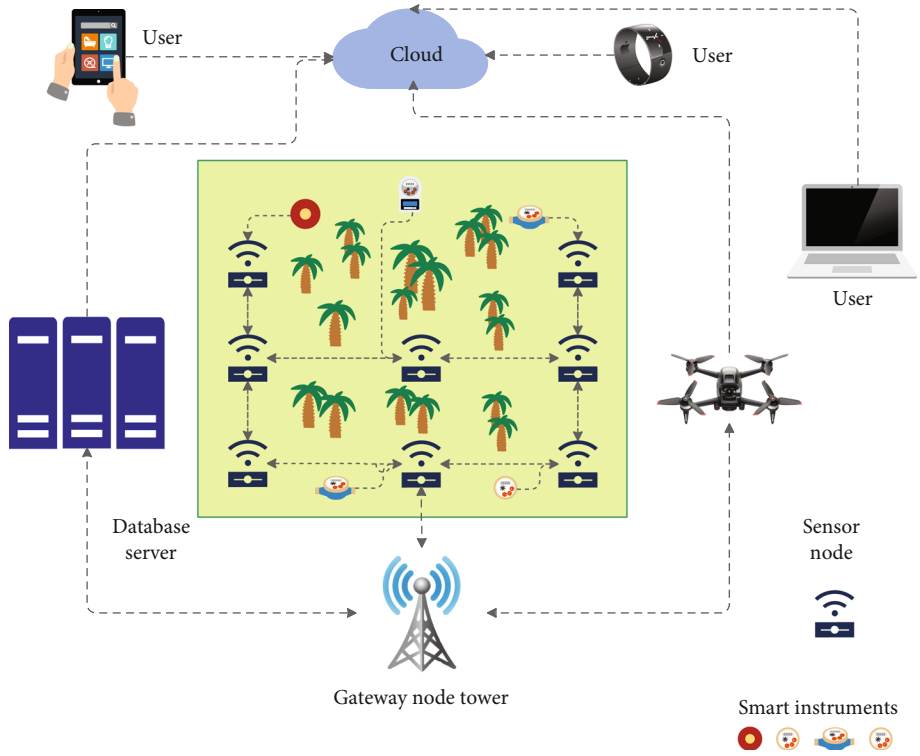

Figure 1: Example of the smart farming application.

every way, smart farming and conventional farming are diametrically opposed. Without regard for market demand, rates, weather predictions, or other variables, traditional farming uses historic and traditional agricultural methods, as well as antiquated equipment for labor and seasonal crop production. Smart farming takes use of modern technology like smart linked devices, Internet of Things sensors, a farmers' chat room, and continuous evaluations of different variables like the optimum circumstances for a plant to develop, the quantity of nutrients needed, soil quality, and water quality monitoring. Smart farming lowers labor costs, boosts crop yields, and enhances production while making farming easy and cheap (cost-effective). Agriculture has progressed to the point where smart farming is the next stage. The use of the Internet of Things (IoT) and unmanned aerial vehicle (UAV) technologies to enhance the efficiency of agricultural operations is referred to as smart farming [3].

A UAV may autonomously and properly reacts to its surroundings based on its context. Agricultural products must be increased mostly because to the rapid expansion of the global population, despite substantial contributions from scientific discoveries in genetics, chemistry, and robotics to the improvement of agricultural technology [4]. At the same time, the agricultural sector is confronted with major challenges such as climate change, land scarcity, and the growing need for freshwater. Information and Communication Technology (ICT) services may be a potential solution to these pressing issues.

UAVs and the IoT are two of the most popular technologies being used for civilian and industrial reasons, as well as to support Industry 4.0. An unmanned aerial vehicle (UAV) is a remotely controlled autonomous vehicle that does not need a human pilot. Unmanned aerial vehicles (UAVs) were originally developed for military purposes, but their growing popularity and technological developments have highlighted their potential for civilian and industrial applications [5, 6]. The Internet of Things (IoT) allows a large number and diversity of linked devices, allowing for remote monitoring and control of various activities [7]. Smart homes or home automation, smart cities, security and surveillance applications, remote patient monitoring, and precision agriculture are all examples of IoT use cases [8–10].

The combination of these two technologies (UAV and IoT) expands the number of options for improving people's lives. Data collection operations in UAV-based applications may be aided by a well-implemented Internet of Things architecture. While UAVs may help gather data from difficult places for IoT applications, the usage of sensor-equipped UAVs in municipal and industrial applications is expanding IoT's power. According to research, UAV-enabled IoT systems might be utilized for a variety of interesting and helpful applications. UAV and IoT integration applications are aimed at smart cities, agriculture, healthcare, disaster management, rescue operations, supply chains, and geoscience [11–13]. Combining UAVs with IoT has a lot of promise, but it also has a lot of technical and legal issues. Examples of applications include air traffic control, obstacle detection, flight schedule and path integrity, the use of different communications designs, data collection via sensors, actual or near real-time data analysis and delivery, and lightweight encryption algorithm to align with restricted onboard resources.

As an emerging technology, blockchain applications are being explored in various industries, including healthcare [14–17], finance [18, 19], real estate [20, 21], agriculture [22, 23], and education [24, 25]. Blockchain implementation is ideal for communication networks, thanks to new improvements in blockchain technology such as decentralization, immutability, security, and transparency. A blockchain is an immutable database that nodes in a distributed and decentralized peer-to-peer network continually update and agree on [26]. Elliptic-curve Public-Key Cryptography is the most prevalent public-key cryptographic technique used in blockchain technology (ECC). This technique has an advantage over Public-Key Cryptography (PKC) in that the authentication and transparency of new transactions are dependent on a widespread agreement among its users. As a result, deploying blockchain technology for distributed UAV networks might provide a slew of security advantages.

The main contribution of this paper is threefold. Firstly, it proposed a novel blockchain-based framework to support Authenticated Wireless Links between a Drone and Sensors. Secondly, it demonstrated a proof-of-concept implementation for the proposed framework by walking through an intelligent farming case study. Lastly, it provided a performance evaluation of the implemented proof of concept.

2. Related Works

The UAV networks' drones can communicate with one another over a wireless link. UAV networks are vulnerable to forgery attacks, man-in-the-middle attacks, and reply to assaults due to their low computing capacity and complicated external environment. Before the drones may communicate with each other, identity authentication is critical, and assuring a legal drone in the network is the top priority of UAV network security. Traditional authentication mechanisms based on username/password or dynamic key have a low level of security. RSA certification necessitates the use of a lengthy session key, which is incompatible with the lightweight requirements of UAV networks. Many security issues are avoided by blockchain's decentralization and secure communications using public cryptography.

This research [9] proposed VAHAK, an Ethereum blockchain-based secure outdoor healthcare medical supply using UAVs. VAHAK enables timely delivery of important medical supplies to critical patients by facilitating decentralized communication between UAVs and entities. In VAHAK, the Ethereum smart contract was utilized to address concerns about security, privacy, and dependability, while the IPFS protocol was used to address storage costs. The VAHAK's security vulnerabilities are tested using the open-source application MyThril. VAHAK is cost-effective in terms of data storage since it uses the InterPlanetary File System (IPFS) for healthcare record storage and 5G-enabled Tactile Internet (TI) for communication. Finally, when compared to conventional systems, VAHAK performance assessment outperformed existing approaches in various performance evaluation parameters such as scalability, latency, and network capacity.

Authors in [27] suggested a blockchain-based intelligent technique for securing the privacy of unmanned aerial vehicles (UAVs) and drones. They presented the hashing process and how it was used in their system, including the creation of a hash code. They created a security mechanism that encrypts information using hashing by combining picture collection and sensing from drones and UAVs with blockchain security. All transactions between the server and the drone, as well as the drone's GPS position, were tracked using the timestamp.

Researchers in [28] built a blockchain-based access management system for the IoD environment, which allows for secure communication between drones and the GSS. Secure data is collected by the GSS in the form of transactions, which are subsequently converted into blocks. Finally, in a peer-to-peer cloud server network, cloud servers connected to the GSS through the Ripple Protocol Consensus Algorithm (RPCA) upload the blocks to the blockchain. After they have been added to the blockchain, the transactions in the blocks cannot be modified, edited, or even removed. They carried out several security analyses, including formal security under the random oracle model, informal security, and simulation-based formal security verification, to ensure that the proposed scheme can withstand a wide range of potential attacks with a high probability, as is required in an IoD environment.

According to Khalifeh et al. [29], a UAV might be used as a data mule to unload sensor nodes and securely transfer monitoring data to a remote control center for further analysis and decision-making. They also spoke about the challenges of putting the proposed framework into reality. Experimenting with their suggested design in the presence of different types of obstructions may be found in typical outdoor fields. During the testing, some differences between the performance metrics provided in the hardware-specific datasheets were uncovered. They uncovered disparities between the declared coverage distance and signal strength via their experiments.

A study in [30] employed a blockchain-enabled identity authentication scheme and a safe data sharing paradigm for drones. Authentication and access control are handled by smart contracts, account creation and security are handled by Public-Key Cryptography, and security auditing is handled by a distributed ledger. To speed up outsourced calculations, ABEM-POC, which is based on the Spark cluster and MapReduce architecture, is presented. The ABE and a modified approach based on ABEM-POC can be used to facilitate parallel outsourced computations. The results showed that both the ABEM-POC and general techniques were successful and straightforward to implement.

ACSUD-IoD, an access control approach for illegal UAV detection and mitigation in an IoD environment, was presented by the authors in [31]. The transactional data to the GSS was stored on a private blockchain, allowing the GSS to identify unauthorized UAVs. They used a range of security tests to show that the suggested system is resilient to a variety of potential attacks that may occur in an IoD environment. Many cryptographic primitives' effectiveness and resilience have been shown in trials.

This study in [32] offered a novel approach for safeguarding communications between unmanned aerial vehicles (UAVs) engaged in various tasks. A one-of-a-kind method for UAVs to enable network transactions without delivering encrypted communications is included in the proposed technique. They also proposed a consensus method based on the proof of communication. They concluded that the suggested approach may be used safely in communication networks.

To mitigate such attacks and achieve trust, this paper [33] proposed a new and systematic framework that combines interest-key-content binding (IKCB), forwarding strategy, and on-demand verification to investigate poisoned content quickly and effectively in NDN-based unmanned aerial vehicles ad hoc networks (UAANETs). They presented a permissioned blockchain network built on top of NDN, as well as a scalable adaptive delegate consensus approach for providing a decentralized IKCB store and detecting internal attackers. Their results show that the suggested architecture may effectively cleanse poisoned material at a low cost and that their techniques performed well enough to be suitable for UAANETs.

In this research [34], the SENTINEL architecture was proposed to facilitate mutual authentication between drones and base stations. SENTINEL generates a flight session key for a drone with a flight plan and registers the flight session key and the drone's flight plan in a centralized database accessible by all ground stations. Ground stations utilize the registered flight session key to authenticate the drone as the MAC key when it is flying. We devised a straightforward certificate format that may be utilized in IoD scenarios. The proposed certificate is designed to convey just the bare minimum of data required to construct public key infrastructure in the Internet of Things (IoT) scenarios. To reduce certificate size even further, they chose a binary format instead of the human-readable text format used in X.509 v3 certificates.

A unique task-oriented authentication method based on blockchain (ToAM) for UAVs was proposed in [35]. They divided UAV authentication into group building authentication and intragroup authentication using a two-stage authentication architecture. They also exhibited a lightweight and cross-domain authentication system based on blockchain that enables for the secure purchase of cross-domain UAVs and task group setup. The job is then performed utilizing a chord ring and a preshared key authentication protocol, which allows for quick and secure authentication inside the UAV group even when the network connection is poor.

The authors of [36] presented a mutual-healing group key distribution technique based on blockchain. The GCS group keys are stored on a private blockchain that is integrated into the Ground Control Station (GCS). Meanwhile, the blockchain was used to manage a dynamic list of UAANET membership certificates. According to different attack situations, a basic mutual-healing protocol and an upgraded one were designed based on the Longest-Lost-Chain approach to recover the node's lost group keys with the aid of its neighbors.

AKMS-AgriIoT, a private blockchain-based system (IPA) for Intelligent Precision Agriculture, was recommended in this study [37]. To confirm and add the blocks created by the encrypted transactions and their accompanying signatures by the GSS to the private blockchain center, the cloud servers mine them. According to extensive security analysis and comparative study, the recommended AKMS-AgriIoT was also given. Table 1 presents a comparison between the existing literature and the proposed framework.

3. Materials and Methods

3.1. Elliptic Curve Cryptographic Protocols (ECC). ECC is a modern family of public-key cryptosystems based on the algebraic structures of elliptic curves over finite fields and the Elliptic Curve Discrete Logarithm Problem's difficulty (ECDLP) [38–40]. ECC provides asymmetric cryptosystem features such as encryption, signatures, and key exchange. ECC is a logical successor to the RSA cryptosystem since it requires fewer keys and signatures to provide the same degree of security as RSA and allows for very fast key generation, key agreement, and signatures. In the ECC, the private keys are integers (typically 256-bit integers in the field size range of the curve). In ECC cryptography, key generation is as easy as dependably generating a random integer inside a given range, making it very quick. A valid ECC private key is an integer within the range.

The ECC's public keys are the EC points, which are pairs of integer coordinates x, y that fall on the curve. Because of their unique properties, EC points may be reduced to only one coordinate plus one bit (odd or even). As a result, the compressed public key is a 257-bit integer, which is equivalent to a 256-bit ECC private key. An ECC public key is an example (corresponding to the above private key, encoded in the Ethereum format, as hex with prefix 02 or 03). The public key needs 33 bytes (66 hex digits) in this format, which may be lowered to roughly 257 bits.

The PKC is a method of generating a pair of keys: public keys that are widely distributed and private keys that are only known by the authorized owner. This serves two purposes: authentication (the public key confirms that the message was sent by the owner of the private key) and encryption (the message can only be decoded by the owner of the associated private key). In this section, ECC has been used for PKC since it provides stronger security with shorter computation time than DSA and RSA. The purpose of Pseudocode 1 is to generate public and private key pairs for authorized users while also sharing data from the smart contract. The computation time for key creation is the same as for symmetric cryptography, but it provides data security and authentication. In the next asymmetric key algorithm of our proposal, we uses private key and public key (private key to P_r key, public key to P_u) [41].

As shown in Pseudocode 2, the $User_A$ encrypt message 'M' by the $User_B$ public key p_B, so that only authorized $User_B$ can decrypt the message. $User_A$ encrypts message 'M' using the $User_B$ public key P_B, as described in Pseudocode 2, so that only authorized $User_B$ can decode the message.

Table 1: A comparison between existing literature and the proposed framework.

Article ref.	Aerial platform	Hash function cryptography	Blockchain-based	System evaluated?	Security framework	Contribution	Issues
[28]	✓	✗	✗	✗	RPCA protocol	Link between the GSS and cloud servers	High-cost complex
[30]	✓	✗	✓	✗	Public key	ABEM-POC	Retrieve key High cost
[31]	✓	✗	✓	✗	Private key	ACSUD-IoD	High-cost complex
[32]	✓	✓	✓	✗	Public key, OTP encryption	Decentralized securing communications	Not able to authenticate in weak connections
[33]	✓	✗	✗	✗	Systematic key	SDN and NFV-based fleet	Authentication in weak connections
[34]	✓	✗	✗	✗	Public key	Binary format, SENTINEL produces a flight session key	Complex, retrieve key, cost
[36]	✓	✗	✗	✗	Private key	Longest-Lost-Chain method, dynamic list certificates	Complex
[37]	✓	✗	✓	✗	Private key	AKMS-AgriIoT + AI	Cloud server cost
Proposed framework	✓	✓	✓	✓	ECC asymmetric key	SHA256 hashing + IPFS data using a smart contract	Complex

Input: E_q (a, b), G, q
Output: generate P_r key, P_u key

1. Eq (a, b): The parameter of ECC that include a,b,q where q is a prime number and form of 2^m
2. G: specific point on curve whose order is big value 'n'.
3. generate $User_A$ key and Pr key select n_A; $n_A < n$, where n is limitation of curve point.
4. Calculate P_u key p_A; $p_A = n_A * G$
5. $User_B$ key generate, and Pr key select n_B; $n_B < n$, where n is limit of curve point.
6. P_u keys calculate p_B; $p_B = n_B * G$
7. $User_A$ key calculate $K_A = n_A * p_B$
8. UserB key calculate KB = nB ∗ pA

Pseudocode 1: Generate ECC asymmetric Key exchange.

Next steps made by the $User_A$: -1. Suppose message be 'M'
2. Encoding the message 'M' into a point on the elliptic curve
3. Let the point be Pm
4. Choose a random + integer 'K' to encrypt the point
5. Cm =K∗G∗Pm +KpB where G is the base point.

Pseudocode 2: Encrypting data using ECC.

The $User_B$ recipient does next steps: -1. Multiplication between first point in the pair with $User_B$ secret key
2. Compute $K_B * G * n_B$
3. subtraction it from second point in the pair
$P_m + K * p_B - (K * G * n_B)$
$P_m + K * p_B - (K * p_B) = P_m$, where $[n_B * G = p_B]$

Pseudocode 3: Decrypting data using ECC.

0x428a2f98	0x71374491	0xb5c0fbcf	0xe9b5dba5	0x3956c25b	0x59f111f1	0x923f82a4	0xab1c5ed5
0xd807aa98	0x12835b01	0x243185be	0x550c7dc3	0x72be5d74	0x80deb1fe	0x9bdc06a7	0xc19bf174
0xe49b69c1	0xefbe4786	0x0fc19dc6	0x240ca1cc	0x2de92c6f	0x4a7484aa	0x5cb0a9dc	0x76f988da
0x983e5152	0xa831c66d	0xb00327c8	0xbf597fc7	0xc6e00bf3	0xd5a79147	0x06ca6351	0x14292967
0x27b70a85	0x2e1b2138	0x4d2c6dfc	0x53380d13	0x650a7354	0x766a0abb	0x81c2c92e	0x92722c85
0xa2bfe8a1	0xa81a664b	0xc24b8b70	0xc76c51a3	0xd192e819	0xd6990624	0xf40e3585	0x106aa070
0x19a4c116	0x1e376c08	0x2748774c	0x34b0bcb5	0x391c0cb3	0x4ed8aa4a	0x5b9cca4f	0x682e6ff3
0x748f82ee	0x78a5636f	0x84c87814	0x8cc70208	0x90befffa	0xa4506ceb	0xbef9a3f7	0xc67178f2

Figure 2: A hash function mechanism with length of 256 bits.

In Pseudocode 3 [41], the message was decrypted by $User_B$ using the private key K_B. Because of PKC's secret key mechanism, the message's originality cannot be tampered with.

3.2. Hash Function Cryptography. Several hash functions are widely used in different applications, including MD5, SHA-160, and SHA-256. The MD5 produces a 128-bit hash value, whereas the SHA-160 and SHA-256 produce a 160-bit and a 256-bit hash value, respectively. Some hash functions have demonstrated weaknesses throughout further research, though all are considered adequate for noncryptographic applications. For instance, vulnerabilities were discovered in MD5, and it is no longer recommended for cryptographic applications, but it is still used to validate file transfers and database partitioning [42]. Similarly, vulnerabilities were discovered in SHA-160, which is no longer recommended for cryptographic applications [43]. On the other hand, the SHA-256 is recommended by the National Institute of Standards and Technology (NIST) to use instead of MD5 or SHA-160 for cryptographic applications [44].

SHA-256 (secured hashing, FIPS 182-2) is a 256-bit digest cryptographic algorithm. It is an MDC or a unique hash function (Manipulation Detection Code) [45]. A message is broken down into $512 = 16 \times 32$-bit blocks, with each block taking 64 rounds [46, 47]. The 32 initial bits of the fractions portions of the cube roots of the first 64 prime integers gives us the 64 binary characters Ki as shown in Figure 2.

4. The Proposed Framework

4.1. Framework Design. There are many different interpretations of security. Confidentiality, which prevents unauthorized release of information, integrity, which prevents illegal change or deleting data, and availability, which prevents unauthorized withhold of data, make up security [48].

One of the most important aspects of any information system is data integrity. Protecting data from illegal alteration, deletion, or fabrication is known as data integrity. Managing an entity's access and rights to certain corporate resources helps to guarantee that sensitive data and services are not misused, misappropriated, or stolen. Authorization is a method of restricting data access. It is the method through which a system determines what level of access a certain authorized user should have to the system's secure resources. To establish a solid cryptographic authentication, the authentication technique we used here is an asymmetric key exchange with an ECC authentication algorithm and SHA-256 hashed data within a smart contract. Because the data is hashed using the proposed SHA-256 method before being stored inside a blockchain, this approach ensures data integrity benefits because users can compute the hash data each time, they want to retrieve hashed data.

The proposed model consists of two parts on-chain components and off-chain components to guarantee data availability between users. The components of mentioned On-chain are a smart contract and blockchain. The Interplanetary File System (IPFS) (https://ipfs.io/) is a system that allows IPFS which is a peer-to-peer file-sharing system that authenticates and transports data using cryptographic hash functions.

The primary goal of IPFS is to efficiently store large files. When storing private or secret data on the cloud, data confidentiality is critical. A data confidentiality, authentication, and access control might be solved by improving cloud reliability and trustworthiness so that we propose an on-chain components based blockchain and off-chain components based IPFS. It makes use of a distributed architecture-based file storage system in which each server may save a fraction of the complete data, resulting in a reliable file storage and sharing system. IPFS uses content addresses to name files. Signals, photos, and any other types of data can be saved on an IPFS server in a system context. Figure 3 describes the transfer process between on-chain and off-chain components and presents the communication between user's keys from different users to transmit read/write secured data procedures. An off-chain component is presented to minimize the cost of the model and guarantee the model privacy. To build off-chain storage to our proposal, an IPFS for key management data is presented to make a decentralized system more secured. The off-chain components The off-chain components is commonly used forfor data storage and to ease an increase the communication simplicity. The proposed EEC cryptographic algorithm is used for secure authentication process between sensors, and drones. All user's data are stored in the IPFS data to let the model be integrated and lightweight as the blockchain cannot store a lot of data.

Then smart contract is created and stores all users' cryptographic keys then connected with blockchain as on-chain components stage using Solidity Language for creating the proposed smart contract. Solidity is a high-level object-oriented language for creating smart contracts. Smart contracts are well-known software that able to control of how accounts behave in the Ethereum state.

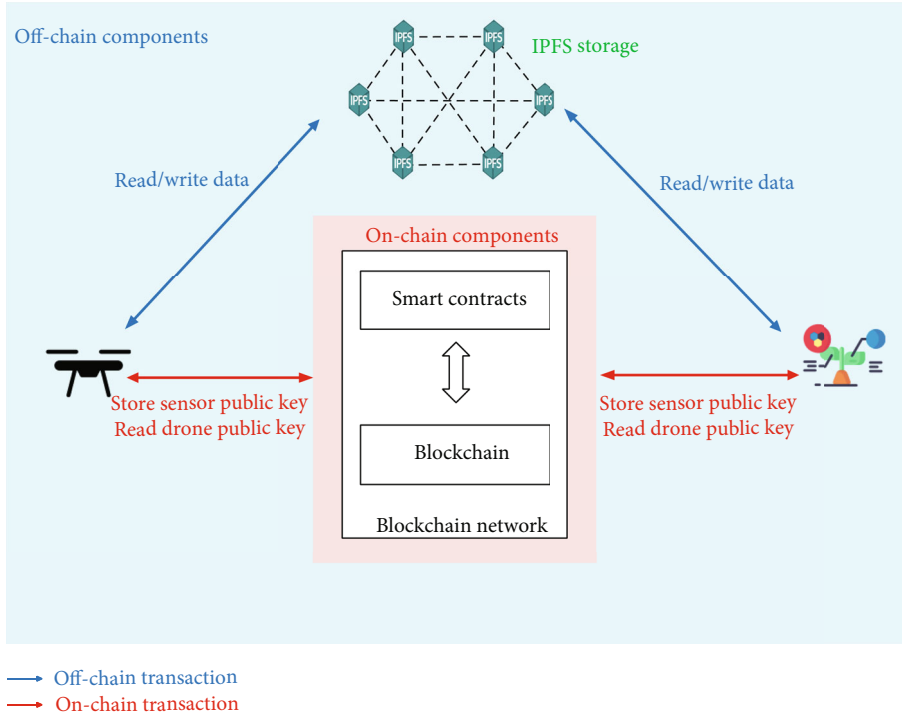

Figure 3: A general diagram of the proposed framework.

Table 2: System's smart contracts main functions.

#	Function	Descriptions
1	*registerNewUser*	This function is responsible for registering a new users in the system
2	*requestData*	This function is responsible for requesting data from a specific user in the system
3	*provideData*	This function is responsible for storing requested data

4.2. Smart Contract. The system smart contract contains three main functions namely *registerNewUser*, *requestData*, and *provideData*. Table 2 provides a description of each function in the smart contract.

4.2.1. The registerNewUser Function. Algorithm 1 describes the process of registering new system users. This function is executed by the user passing the relevant information, including user wallet address, user public key, and role. Based on the user role, smart contracts store user information in the relevant on-chain storage. The smart contract then notifies the system admin, who is responsible for setting up the system, to validate users' information through an off-chain process. After successful validation, the admin executes a specific smart contract function to approve the user registration request.

4.2.2. The registerNewUser Function. Algorithm 2 describes the process of requesting data from a specific user in the system. The function utilizes a mapping data structure for efficient data storing and retrieval. The request information includes the request identification number, sensor wallet address, and drone wallet address. When request information is stored, the smart contract notifies the relevant user to process the request.

Input: wallet, publicKey, role
Output: response
1 User⟵*mapping*
2 **if** *User*[*wallet*].*wallet* == *null* **then**
3 User.insert(wallet, [wallet, publicKey, role])
4 response:*successful*
5 **else**
6 response: revert smart contract state

Algorithm 1: registerNewUser.

4.2.3. The provideData Function. Algorithm 3 describes the process of providing requested data. To provide data for a specific request, the relevant user, the data owner, prepares the requested data and then executes this function, passing the request identification number and the data. When requested data is available, the smart contract notifies the relevant user to retrieve the data.

5. Results

5.1. A Proof of Concept. We implemented a proof of concept to demonstrate the feasibility of the proposed framework. We used Hyperledger Besu (https://www.hyperledger.org/

Input: id, semsorWallet, dronerWallet
Output: response
1 Request⟵*mapping*
2 **if** *User*[*sensorWallet*].*role* == *Sensor* AND *User*[*dronerWallet*].*role* == *droner* **then**
3 Request.insert(*id*,[sensorWallet, dronerWallet])
4 response: *successful*
5 **else**
6 response: revert smart contract state

Algorithm 2: : requestData

Input: id, sensorWallet, encryptedData
Output: response
1 Request⟵*mapping*
2 **if** *User*[*sensorWallet*].*role* == *Sensor* AND *User*[*dronerWallet*].*role* == *droner* **then**
3 Request.insert(*id*, [encryptedData])
4 response: *successful*
5 **else**
6 response: revert smart contract state

Algorithm 3: provideData.

```
pragma solidity 0.5.16;
pragma experimental ABIEncoderV2;

contract BUN {
    // User roles in the system
    enum ROLE {
        Null,
        Admin,
        Sensor,
        Drone
    }

    // data storage for user profile
    mapping(address => User) public user;
    mapping(uint256 => Request) public request;

    address[] public userIds;
    uint256[] public requestIds;
    uint256 lastrequestId;

    struct User {
        address wallet;
        bytes publicKey;
        ROLE role;
    }

    struct Request {
```

Figure 4: A screenshot of the system smart contract.

```
Contract: BUN
  Smart Contract Deployment
    ✓ should set admin
  New user registration
    ✓ should register a new Sensor (80ms)
    ✓ should register a new Drone (71ms)
    ✓ should NOT register existing user (115ms)
    ✓ only admin can add new  user
  Request Data
    ✓ should send data request (62ms)
    ✓ should NOT send data request if Sensor is not registered before
    ✓ should NOT send data request if caller is not Drone
  Provide Data
    ✓ should provide data (42ms)
    ✓ should NOT provide data if data provided before (72ms)
    ✓ should NOT provide data if caller is not the same Sensor (82ms)
    ✓ should NOT provide data if caller is not Sensor
  Reading data
    ✓ get users information
    ✓ get requests information
    ✓ get user Public Key value


15 passing (4s)
```

Figure 5: The result of system smart contract testing.

use/besu) to build a permissioned blockchain. The system smart contract was written using the Solidity programming language, where the Truffle framework (https://www.trufflesuite.com/truffle), an Ethereum smart contracts development tool, was used to test, compile, and deploy the system smart contract. In Figure 4, a screenshot of the system smart contract shows a screenshot of the system smart contract, whereas in Figure 5, the result of system smart contract testing shows the result of smart contract testing undertaken. Lastly, we utilized Node.js and IPFS to develop the off-chain components. The source code is available on Mendeley Data [49] under the CC BY 4.0 license.

The high-level structure of the implemented proof of concept is shown in Figures 6–8. All users in the network save their public keys in the smart contract and the read/write procedures for different user's keys from drones and sensors as explained in Figure 4, for example, a user 1 who wants to send data to user 2 considering the next steps:

(i) User 1, send data to user 2

(a) The data of the user 1 form is hashed using a hash function, which is passed to the hash value and then stored by smart contract in the blockchain

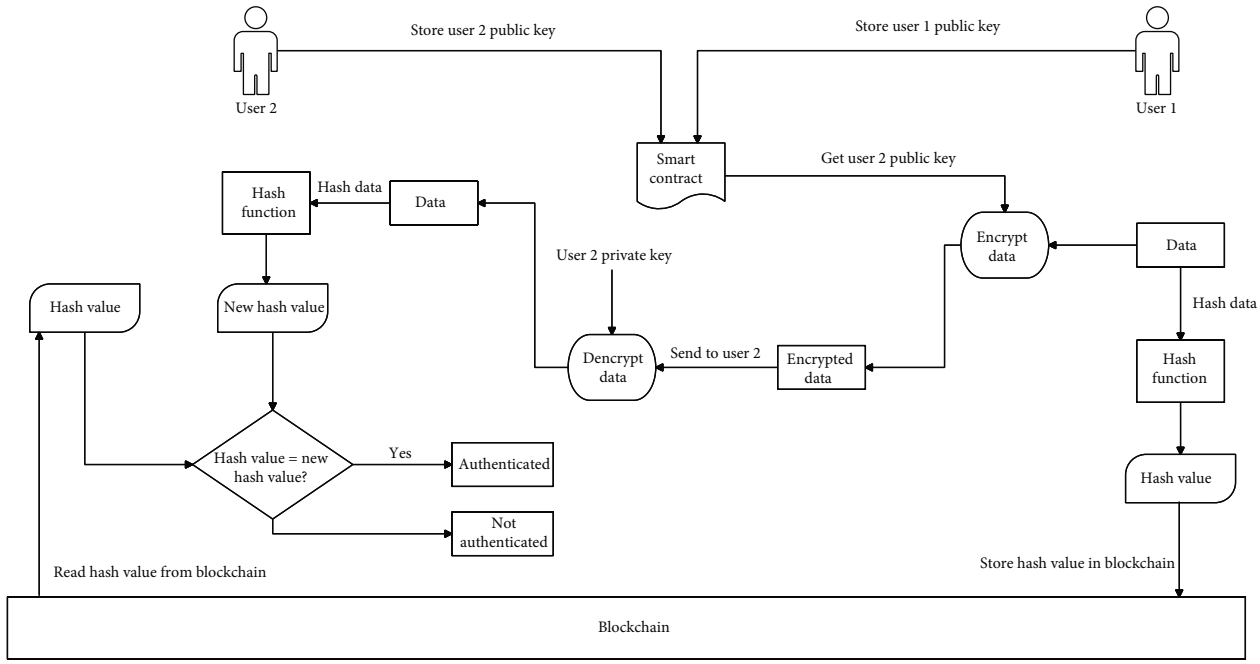

Figure 6: The high-level structure of the proof of concept.

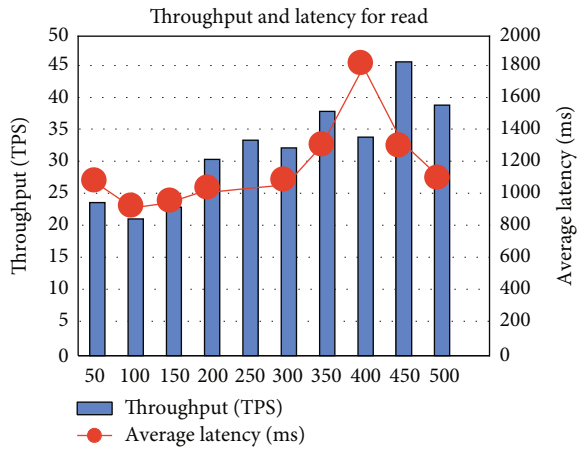

Figure 7: The *throughput* and *latency* results for Read operations.

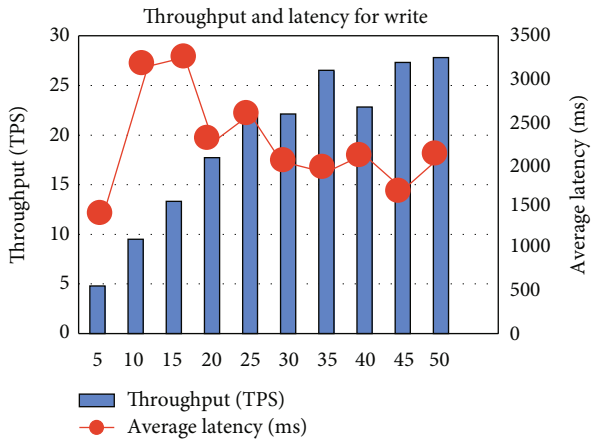

Figure 8: The *throughput* and *latency* results for Write operations.

(ii) User 2 receives data

(a) The data is encrypted using user 2's public key, which is read from user 2's smart contract, and this data is then sent to user 2

(b) With user 2's private key, the data is decrypted, and that data is then hashed into a new hash value

(c) Decision-making occurs when a match is made between two hash values: the hash value is saved in the blockchain and the new hash value. If the hash is matched, authentication occurs, and data is sent

5.2. Performance Evaluation. To evaluate the performance of proposed framework, we utilized an open-source benchmarking tool called Hyperledger Caliper (https://www.hyperledger.org/use/caliper). Table 3 shows the settings of the performance evaluation environment. Two main types of blockchain operations, *Write* and *Read*, were evaluated using four performance indicators, namely, *Write Throughput*, *Read Throughput*, *Write Latency*, and *Read Latency* [50, 51]. In this performance evaluation, we focused on the main smart contract functions that are shown in Table 4.

The performance evaluation was performed in ten rounds with a hundred transactions per round to reduce the likelihood of errors due to network congestion and system overload. Tables 5 and 6 summarize the performance evaluation settings used for the Write and Read operations, respectively.

Table 7 demonstrates the results of the *throughput* and *latency* for Read operations and the *throughput* and *latency*

Table 3: The settings of the performance evaluation environment.

Factor	Setting
Nodes	Four VMs running on Google cloud, where each VM has a 2 GHz 4-core Intel CPU
Peer-to-peer network	Hyperledger Besu v1.4.1, 1 validator node, 3 peer nodes
Consensus protocol	Clique
Smart contract programming language used	Solidity
Benchmarking tool	Hyperledger Caliper v0.4.1

Table 4: Main smart contracts functions of the system.

Main function	Operation type
registerNewUser	*Write*
requestData	*Write*
provideData	*Write*
getUserPublicKey	*Read*
getUsers	*Read*
getRequests	*Read*

Table 5: The performance evaluation settings used for the Write operations.

Test number	1	2	3	4	5	6	7	8	9	10
Functions under test	Write operations									
Worker number	1 worker									
Transaction number	100 transactions									
Type of control rate	Fixed rate									
Send rate (tps)	5	10	15	20	25	30	35	40	45	50

for Write operations, respectively. The results indicate an average *throughput* of 32.54 TPS and an average latency 1166 milliseconds for Read operations. Contrastingly, average Write *throughput* is 19.37 TPS, and the average Write *latency* is 2253 milliseconds. The experimental findings for the *Read* and *Write* operations are shown in Tables 7 and 8.

5.3. Discussion. Blockchain technology has developed as a way to make distributed systems more secure so that we provide security to the network users confidential data over the cloud. Blockchains are digital ledgers that hold explicit and verifiable records of all transactions inside a system. The decentralized blockchain concept has shown to be a reliable technique for resolving trust difficulties in user authentication. As a result of this fact, the specifics of each transaction could be saved in order to ensure that the data transmission is secure. The major objective of this work is to guarantee that the system's authentication is robust and safe against assaults, since each user has their own private and public keys, which were previously issued to them by the system and stored on the smart contract.

To establish a solid cryptographic authentication, the technique uses an asymmetric key exchange with an ECC authentication algorithm and SHA-256 hashed data within a smart contract. The model is built on a private blockchain-based platform that ensures safe connection and secure data transmission through the cloud between sensors and drones in smart farming. This work supports our system by ensuring data integrity since data is hashed before being recorded in a blockchain, and users calculate the hash data each time they want to access hashed data.

The proposed framework utilizes permissioned blockchain where only authorized users can access the system. System user interacts with the system using their wallet accounts which are pseudo-anonymous accounts; therefore, users' privacy is preserved. In addition, multiple pseudo-anonymous accounts can be used by a single user; hence, user transactions cannot be tracked by an adversary. The use of on-chain/off-chain storage in the framework increases data confidentiality and integrity as sensitive data are stored securely off-chain and only the hash value of the data is submitted to the blockchain.

The major goal of this work is to guarantee that the system's authentication is stable and safe against assaults, since each user has their own private and public keys, which were previously issued by the system and stored on the smart contract, before registering and entering the system. To establish a solid cryptographic authentication, the work is based on an asymmetric key exchange within a smart contract utilizing an ECC authentication algorithm and SHA-256 hashed data. The system is built on a private blockchain-based platform that ensures safe connection and secure data transmission through the cloud between sensors and drones in smart farming. Because the data is hashed before being stored within a blockchain, this method protects data integrity, and users may calculate the hash data each time they want to access hashed data.

6. Conclusions

We proposed an asymmetric key cryptography blockchain as an on-chain component for this study, which requires storing data on permission blockchain as the most cost-effective way to keep data decentralized and guarantee model availability. We looked at smart contracts on the blockchain that can be utilized in the realm of the Internet of Things, as well as the benefits and challenges that they bring. We used Ethereum smart contracts, a decentralized and encrypted technology that allows devices to better trust one another and execute peer-to-peer authentication.

The hashed data will simply be transmitted on the on-chain component. No one will be able to access the model's private keys after they are set, which validates the model's privacy.

We used an Ethereum blockchain to test the performance of the proposed approach. Four virtual computers

Table 6: The performance evaluation settings used for the Read operations.

Test number	1	2	3	4	5	6	7	8	9	10
Functions under test	Read operations									
Worker number	1 worker									
Transaction number	100 transactions									
Type of control rate	Fixed rate									
Send rate (tps)	50	100	150	200	250	300	350	400	450	500

Table 7: Experimental findings of *throughput* and *latency* for Read operations.

Test round	Send rate (tps)	Average latency (ms)	Throughput (TPS)
1	50	1050	23.9
2	100	920	21.3
3	150	950	23.2
4	200	1020	30.4
5	250	1040	33.7
6	300	1060	32.6
7	350	1350	40.8
8	400	1800	34.4
9	450	1330	45.7
10	500	1140	39.4

Table 8: Experimental findings of *throughput* and *latency* for Write operations.

Test round	Send rate (tps)	Average latency (ms)	Throughput (WPS)
1	5	1420	4.8
2	10	3180	9.5
3	15	3260	13.3
4	20	2240	17.7
5	25	2500	21.9
6	30	2040	22.1
7	35	1960	26.5
8	40	2100	22.8
9	45	1710	27.3
10	50	2120	27.8

from the Google cloud are used to guarantee the newly added device's security needs while also achieving benefits such as reduced traffic overheads and typifying our solution's high level of intelligence and mobility. We believe that the blockchain solution is a step toward greater data security and privacy and that it has the potential to be employed in a wide range of IoT applications.

This study has several limitations. In the proposed framework, verifying user identity is challenging due to the distributed and the openness nature of blockchain technology. As the proposed framework operates on a permissioned blockchain, the process of verifying user identity is performed by the system owner, who is responsible for setting up the blockchain and inviting users to join the system. This can be mitigated by integrating the system with identity management services such as self-sovereign identity [52–54], identity verification using blockchain [55], and noncustodial login solutions using blockchain [56]. The blockchain's General Data Protection Regulation (GDPR) compliance is another limitation in this study [57–59]. Although utilizing permissioned blockchains might comply with GDPR requirements, determining whether blockchain completely complies with GDPR is challenging [60]. To avoid such limitation, GDPR compliance should be considered during designing the blockchain-based system [61, 62].

Throughout this study, we described an approach on how to achieve trust among smart farming users. We now highlight some future research directions. Firstly, we will improve our work by incorporating an off-chain (IPFS)-based decentralized distributed data storage method to allow for speedy, low-cost, and reliable data access, hence, approving the model's availability. Accessing data across users and networks in a faster, more secure, and network-effective manner is another advantage of using off-chain components. Secondly, a comparative analysis of the proofs of concept implemented with different blockchain frameworks and configurations will be conducted. In this study, the proofs of concept were implemented using the Ethereum blockchain, which was initially designed for developing DApps and services that are open to the public. Ethereum blockchain has several limitations in terms of performance and scalability, such as transaction latency, throughput, and execution time [63]. In future works, an empirical evaluation should be conducted to assess the performance of the proofs-of-concept design under a wide range of blockchain frameworks and configurations other than Ethereum such as Hyperledger Fabric and MultiChain. Finally, more research is required on the framework applicability to explore and assess the sustainability challenges faced by our proposed framework, including its limitations for real-world utilizing.

Data Availability

There are no relevant data to be made available.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] R. Lal, "Feeding 11 billion on 0.5 billion hectare of area under cereal crops," *Food and Energy Security*, vol. 5, no. 4, pp. 239–251, 2016.

[2] M. Z. Mehmood, M. Ahmed, O. Afzal et al., "Internet of Things (IoT) and sensors technologies in smart agriculture: applications, opportunities, and current trends," in *Building Climate Resilience in Agriculture*, pp. 339–364, Springer, 2022.

[3] A. Rehman, T. Saba, M. Kashif, S. M. Fati, S. A. Bahaj, and H. Chaudhry, "A revisit of internet of things technologies for monitoring and control strategies in smart agriculture," *Agronomy*, vol. 12, no. 1, p. 127, 2022.

[4] F. A. Almalki, B. O. Soufiene, S. H. Alsamhi, and H. Sakli, "A low-cost platform for environmental smart farming monitoring system based on IoT and UAVs," *Sustainability*, vol. 13, no. 11, p. 5908, 2021.

[5] F. A. Almalki and B. O. Soufiene, "Modifying Hata-Davidson propagation model for remote sensing in complex environments using a multifactional drone," *Sensors*, vol. 22, no. 5, p. 1786, 2022.

[6] F. A. Almalki and M. C. Angelides, "Autonomous flying IoT: a synergy of machine learning, digital elevation, and 3D structure change detection," *Computer Communications*, vol. 190, pp. 154–165, 2022.

[7] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[8] P. Gope and T. Hwang, "BSN-Care: a secure IoT-based modern healthcare system using body sensor network," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1368–1376, 2015.

[9] H. Arasteh, V. Hosseinnezhad, V. Loia et al., "Iot-based smart cities: a survey," in *2016 IEEE 16th international conference on environment and electrical engineering (EEEIC)*, pp. 1–6, Florence, Italy, 2016.

[10] A. Khanna and S. Kaur, "Evolution of Internet of Things (IoT) and its significant impact in the field of precision agriculture," *Computers and Electronics in Agriculture*, vol. 157, pp. 218–231, 2019.

[11] W. Ejaz, M. A. Azam, S. Saadat, F. Iqbal, and A. Hanan, "Unmanned aerial vehicles enabled IoT platform for disaster management," *Energies*, vol. 12, no. 14, p. 2706, 2019.

[12] A. D. Boursianis, M. S. Papadopoulou, P. Diamantoulakis et al., "Internet of things (IoT) and agricultural unmanned aerial vehicles (UAVs) in smart farming: a comprehensive review," *Internet of Things*, vol. 18, p. 100187, 2022.

[13] F. A. Almalki, M. Aljohani, M. Algethami, and B. O. Soufiene, "Incorporating drone and AI to empower smart journalism via optimizing a propagation model," *Sustainability*, vol. 14, no. 7, p. 3758, 2022.

[14] F. Albalwy, A. Brass, and A. Davies, "A blockchain-based dynamic consent architecture to support clinical genomic data sharing (ConsentChain): proof-of-concept study," *JMIR medical informatics*, vol. 9, no. 11, article e27816, 2021.

[15] F. Albalwy, J. H. McDermott, W. G. Newman, A. Brass, and A. Davies, "A blockchain-based framework to support pharmacogenetic data sharing," *The Pharmacogenomics Journal*, 2022.

[16] U. Chelladurai and S. Pandian, "A novel blockchain based electronic health record automation system for healthcare," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 1, pp. 693–703, 2022.

[17] L. Hang, B. Kim, K. Kim, and D. Kim, "A permissioned blockchain-based clinical trial service platform to improve trial data transparency," *BioMed Research International*, vol. 2021, Article ID 5554487, 22 pages, 2021.

[18] V. S. Anoop and J. Goldston, "Decentralized finance to hybrid finance through blockchain: a case-study of acala and current," *Journal of Banking and Financial Technology*, vol. 6, no. 1, pp. 109–115, 2022.

[19] D. Younus, A. Muayad, and M. Abumandil, "Role of smart contract technology blockchain services in finance and banking systems: concept and core values," *Mohanad, Role of Smart Contract Technology Blockchain Services in Finance and Banking Systems: Concept and Core Values (April 8, 2022)*, 2022.

[20] K. Azari and S. Malek, *Blockchain Applications in Real Estate: Challenges and a Proposed Framework*, 2022.

[21] A. Patil, A. Shinde, A. Panigrahi, A. Arora, D. S. Raviraja, and R. Babu, "The role of blockchain technology in decentralized real estate marketplace: recent findings," 2022.

[22] F. Jamil, M. Ibrahim, I. Ullah, S. Kim, H. K. Kahng, and D.-H. Kim, "Optimal smart contract for autonomous greenhouse environment based on IoT blockchain network in agriculture," *Computers and Electronics in Agriculture*, vol. 192, p. 106573, 2022.

[23] L. Hang, I. Ullah, and D.-H. Kim, "A secure fish farm platform based on blockchain for agriculture data integrity," *Computers and Electronics in Agriculture*, vol. 170, p. 105251, 2020.

[24] M. K. Dash, G. Panda, A. Kumar, and S. Luthra, "Applications of blockchain in government education sector: a comprehensive review and future research potentials," *Journal of Global Operations and Strategic Sourcing*, vol. 15, no. 3, pp. 449–472, 2022.

[25] A. Garg, P. Kumar, M. Madhukar, O. Loyola-González, and M. Kumar, "Blockchain-based online education content ranking," *Education and Information Technologies*, vol. 27, no. 4, pp. 4793–4815, 2022.

[26] H. F. Atlam, A. Alenezi, M. O. Alassafi, and G. Wills, "Blockchain with internet of things: benefits, challenges, and future directions," *International Journal of Intelligent Systems and Applications*, vol. 10, no. 6, pp. 40–48, 2018.

[27] A. Yazdinejad, R. M. Parizi, A. Dehghantanha, H. Karimipour, G. Srivastava, and M. Aledhari, "Enabling drones in the internet of things with decentralized blockchain-based security," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6406–6415, 2020.

[28] B. Bera, D. Chattaraj, and A. K. Das, "Designing secure blockchain-based access control scheme in IoT-enabled Internet of Drones deployment," *Computer Communications*, vol. 153, pp. 229–249, 2020.

[29] A. Khalifeh, K. A. Darabkh, A. M. Khasawneh et al., "Wireless sensor networks for smart cities: network design, implementation and performance evaluation," *Electronics*, vol. 10, no. 2, p. 218, 2021.

[30] C. Feng, K. Yu, A. K. Bashir et al., "Efficient and secure data sharing for 5G flying drones: a blockchain-enabled approach," *IEEE Network*, vol. 35, no. 1, pp. 130–137, 2021.

[31] B. Bera, A. K. Das, and A. K. Sutrala, "Private blockchain-based access control mechanism for unauthorized UAV detection and mitigation in Internet of Drones environment," *Computer Communications*, vol. 166, pp. 91–109, 2021.

[32] E. Ghribi, T. T. Khoei, H. T. Gorji, P. Ranganathan, and N. Kaabouch, "A secure blockchain-based communication approach for UAV networks," in *2020 IEEE International Conference on Electro Information Technology (EIT)*, pp. 411–415, Chicago, IL, USA, 2020.

[33] R. L. Kumar, Q.-V. Pham, F. Khan, M. J. Piran, and K. Dev, "Blockchain for securing aerial communications: potentials, solutions, and research directions," *Physical Communication*, vol. 47, p. 101390, 2021.

[34] G. Cho, J. Cho, S. Hyun, and H. Kim, "SENTINEL: a secure and efficient authentication framework for unmanned aerial vehicles," *Applied Sciences*, vol. 10, no. 9, p. 3149, 2020.

[35] A. Chen, K. Peng, Z. Sha, X. Zhou, Z. Yang, and G. Lu, "ToAM: a task-oriented authentication model for UAVs based on blockchain," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, Article ID 166, 15 pages, 2021.

[36] X. Li, Y. Wang, P. Vijayakumar, D. He, N. Kumar, and J. Ma, "Blockchain-based mutual-healing group key distribution scheme in unmanned aerial vehicles ad-hoc network," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11309–11322, 2019.

[37] B. Bera, A. Vangala, A. K. Das, P. Lorenz, and M. K. Khan, "Private blockchain-envisioned drones-assisted authentication scheme in IoT- enabled agricultural environment," *Computer Standards & Interfaces*, vol. 80, p. 103567, 2022.

[38] D. Maldonado-Ruiz, J. Torres, and N. El Madhoun, "3BI-ECC: a decentralized identity framework based on blockchain technology and elliptic curve cryptography," in *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*, pp. 45-46, Paris, France, 2020.

[39] H. Wang, D. He, and Y. Ji, "Designated-verifier proof of assets for bitcoin exchange using elliptic curve cryptography," *Future Generation Computer Systems*, vol. 107, pp. 854–862, 2020.

[40] A. K. Yadav, "Significance of elliptic curve cryptography in blockchain IoT with comparative analysis of RSA algorithm," in *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pp. 256–262, Greater Noida, India, 2021.

[41] R. Kumar and R. Tripathi, "Secure healthcare framework using blockchain and public key cryptography," in *Blockchain Cybersecurity, Trust and Privacy*, pp. 185–202, Springer, 2020.

[42] Z. E. Rasjid, B. Soewito, G. Witjaksono, and E. Abdurachman, "A review of collisions in cryptographic hash function used in digital forensic tools," *Procedia computer science*, vol. 116, pp. 381–392, 2017.

[43] S. Soni and S. P. Singh, "Secure and efficient integrity algorithm based on existing SHA algorithms," *International Journal of Computer Applications*, vol. 113, no. 11, pp. 34–37, 2015.

[44] NIST Policy on Hash FunctionsJanuary 2022, https://csrc.nist.gov/Projects/Hash-Functions/NIST-Policy-on-Hash-Functions.

[45] W. L. Harrison, A. M. Procter, and G. Allwein, "Model-driven design & synthesis of the SHA-256 cryptographic hash function in rewire," in *2016 International Symposium on Rapid System Prototyping (RSP)*, pp. 1–7, Pittsburgh, PA, USA, 2016.

[46] M. Qazi, D. Kulkarni, and M. Nagori, "Proof of authenticity-based electronic medical records storage on blockchain," in *Smart Trends in Computing and Communications*, pp. 297–306, Springer, 2020.

[47] K. Quist-Aphetsi and H. Blankson, "A hybrid data logging system using cryptographic hash blocks based on SHA-256 and MD5 for water treatment plant and distribution line," in *2019 International Conference on Cyber Security and Internet of Things (ICSIoT)*, pp. 15–18, Accra, Ghana, 2019.

[48] A. Ali, M. F. Pasha, J. Ali et al., "Deep learning based homomorphic secure search-able encryption for keyword search in blockchain healthcare system: a novel approach to cryptography," *Sensors*, vol. 22, no. 2, p. 528, 2022.

[49] F. Albalwy, *blockchain-for-uav-networks*, Mendeley Data, 2022.

[50] Z. Lu, Q. Wang, G. Qu, and Z. Liu, "BARS: a blockchain-based anonymous reputation system for trust management in VANETs," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 98–103, New York, NY, USA, 2018.

[51] Hyperledger, *Hyperledger Blockchain Performance Metrics White Paper*, Hyperledger, 2022, January 2022, https://www.hyperledger.org/learn/publications/blockchain-performance-metrics.

[52] N. Naik and P. Jenkins, "Sovrin Network for decentralized digital identity: analysing a self-sovereign identity system based on distributed ledger technology," in *2021 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 1–7, Vienna, Austria, 2021.

[53] T. Rathee and P. Singh, "A self-sovereign identity management system using blockchain," in *Cyber Security and Digital Forensics*, pp. 371–379, Springer, 2022.

[54] M. Shuaib, N. H. Hassan, S. Usman et al., "Self-sovereign identity solution for blockchain-based land registry system: a comparison," *Mobile Information Systems*, vol. 2022, Article ID 8930472, 17 pages, 2022.

[55] blockpassAugust 2022, https://www.blockpass.org/.

[56] RemmeAugust 2022, https://remme.io/.

[57] M. Berberich and M. Steiner, "Practitioner's Corner Blockchain technology and the GDPR – how to reconcile privacy and distributed ledgers?," *European Data Protection Law Review*, vol. 2, no. 3, pp. 422–426, 2016.

[58] A. V. Humbeeck, "The blockchain-GDPR paradox," *Journal of Data Protection & Privacy*, vol. 2, no. 3, pp. 208–212, 2019.

[59] C. Compert, M. Luinetti, and B. Portier, *Blockchain and GDPR: How Blockchain Could Address Five Areas Associated with GDPR Compliance*, IBM Security, 2018, August 2022, https://iapp.org/media/pdf/resource_center/blockchain_and_gdpr.pdf.

[60] M. Finck, *Blockchain and the General Data Protection Regulation: Can Distributed Ledgers Be Squared with European Data Protection Law?(Study No: PE 634.445)*, European Parliament, Brussels, 2019.

[61] N. Eichler, S. Jongerius, G. McMullen, O. Naegele, L. Steininger, and K. Wagner, *Blockchain, Data Protection, and the GDPR*, Blockchain Bundesverband, 2018, August 2022, https://www.crowdfundinsider.com/wp-content/uploads/2018/06/GDPR_Position_Paper_v1.0.pdf.

[62] A. Rose, "GDPR challenges for blockchain technology," *Interactive Entertainment Law Review*, vol. 2, no. 1, pp. 35–41, 2019.

[63] L. Hang, B. Kim, and D. Kim, "A transaction traffic control approach based on fuzzy logic to improve hyperledger fabric performance," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 2032165, 19 pages, 2022.