IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 November 2023, accepted 1 January 2024, date of publication 4 January 2024, date of current version 22 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3349955

APPLIED RESEARCH

A Novel Distributed Authentication of Blockchain Technology Integration in IoT Services

AVISHAEK DEEP[1], ADOLFO PERRUSQUÍA[1], (Member, IEEE), LAMEES ALJABURI[2], SABA AL-RUBAYE[1], (Life Senior Member, IEEE), AND WEISI GUO[1], (Senior Member, IEEE)

[1]School of Aerospace, Transport and Manufacturing, Cranfield University, MK43 0AL Bedford, U.K.
[2]Computer Engineering Department, Near East University, 99138 Nicosia, Cyprus

Corresponding author: Adolfo Perrusquía (adolfo.perrusquia-guzman@cranfield.ac.uk)

ABSTRACT Internet of Things (IoT) is currently playing a major role in how intelligent devices are interconnected and deployed to automate services in transport and smart living sectors. However, IoT is facing challenges in terms of data protection and authentication due to the heterogeneous nature of IoT devices that do not exhibit a central authority. It is crucial to provide secure and trustworthy solutions for the increasing demands of decentralized IoT environments. To this end, this research proposes a novel integration of blockchain-technologies in IoT services to enhance security, data integrity, users privacy, system scalability and interoperability of devices. This is done by leveraging smart contracts to enforce authentication, access control and data exchange mechanisms for IoT devices. The proposed approach is verified by the construction and deployment of a smart contract over the Polygon blockchain network in a simulated real-world IoT scenario. The obtained results show that the proposed approach ensures fast and secure authentication in IoT networks by decreasing the risk of unauthorized access and data tampering.

INDEX TERMS Smart contracts, Internet of Things (IoT), blockchain-technologies, decentralization, authentication.

I. INTRODUCTION

The Internet of Things (IoT) is a revolutionary technology that has enhanced the way in how smart devices and sensors are coherently interconnected to exchange data in an uninterrupted mechanism [1], [2]. This technology has been applied in smart living and transport sectors (e.g, smart cities and healthcare) to create a seamless and efficient flow of data for intelligent decision-making and automation [3], [4]. However, the proliferation of IoT devices has raised concerns in terms of trust and security. Here, traditional IoT architectures are designed with a centralized scheme such that devices that are connected to the IoT architecture can be vulnerable to threats and, in consequence, the data integrity, privacy and trustworthiness are compromised. This is particularly worrying when sensitive data are collected and exchanged in centralized IoT devices.

The associate editor coordinating the review of this manuscript and approving it for publication was M. Anwar Hossain.

Key security requirements have been established in IoT environments to ensure the confidentiality, integrity, and privacy of sensitive data. These requirements include authentication, authorization and access control, data integrity, interoperability, privacy, and identity management [5]. Here, the most severe threats in IoT are attributed to the lack of compliance of the aforementioned requirements [6], [7]. In addition, the absence of standardized protocols hinders the efficient communication, interoperability and data exchange between devices. This can lead to a fragmented and inefficient IoT environment [8].

Scalability plays an important role in the success of IoT systems by scaling up the network capacity, infrastructure, and computational resources. However, as the IoT environment grows in size, it can produce significant delays, scalability issues, and increased costs [9]. Furthermore, several issues can be observed due to the centralized nature of traditional IoT architectures such as: i) single point failures, ii) network congestions, iii) increased latency,

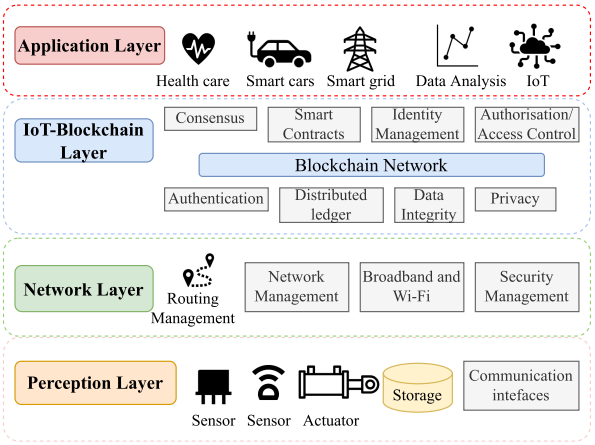


FIGURE 1. Implementation of Blockchain in IoT. Adapted from [14].

and iv) malicious attacks from hackers or unauthorized actors [10], [11].

Blockchain-technology has gained interest in the IoT framework due to its immutability and transparency capabilities in a decentralized architecture. One of the fundamental aspects of IoT is to enable the sharing of resources from constrained devices with other entities. In order to accomplish this, IoT devices must have the capability to oversee and control access to their resources. These requirements can involve various factors, including user requests, technical constraints, and permissions granted to multiple devices, among other considerations. Nevertheless, many IoT devices face limitations in storing and processing the necessary information to independently manage their resources [10], [12]. This challenge often arises due to the constrained nature of these devices.

Blockchain can be classified in public, private, hybrid and consortium depending of its governing factors such as scalability, data privacy, access control and blockchain governance [13]. Pubic blockchain are governed by a decentralized community who make decisions together based on the rules and protocols of the network. Public blockchains suffer a limitation of data privacy since the details of the transactions are visible to all the community. On the other hand, private blockchains are governed by a single entity or organization, that is, their governance is centralized. Here, the central authority determines the access to the network. In contrast to public blockchains, private ones require granted permission to join and verify transactions. Hybrid blockchains combines the benefits of private and public blockchains, that is, they are regulated by a single organization, but rely on the public blockchain for specific verification tasks. Consortium blockchains are governed by a pre-selected group of organizations with common interests known as consortium members. This provides better decentralization capabilities in comparison with private blockchains, which also enhances security and trust and a higher level of scalability and privacy compared with public blockchains.

Fig. 1 depicts a high-level view of the blockchain-IoT integration scheme. The scheme is composed of four layers whose complementary integration enables a secure, transparent and efficient IoT environment [15]. This is done by exploiting the unique features and properties of blockchain to address various security requirements of IoT. These features are:

- The creation of a robust and distributed identity management system [16]. Here, each IoT device will have a unique and immutable identity stored over a distributed ledger [17].
- The implementation of a decentralised authentication mechanisms which means that each IoT device will have a unique digital identifier (ID).
- The use of smart contracts to define authorisation/ access control rules and permissions for IoT devices. This will ensure that only authorized entities can have access to the IoT resources [18].
- Scalability is enhanced due to the decentralized nature of blockchains. This eliminates single points of failure and central authorities [19].
- The transactions and data storing are immutable and cryptographically secured.
- Data is encrypted to ensure sensitive information remains confidential, secure and accessible only to authorized entities.
- The use of smart contracts improves communication, interoperability, security and scalability of IoT environments.

II. RELATED WORK

Blockchain in IoT is an active area for research and deployment. Several platforms have been designed to integrate blockchain in IoT. Some of the most used platforms include IOTA's Tangle, Hdac, VeChain, Streamr, and Waltonchain [20]. A detailed review of blockchain implementation in IoT is given in [21]. Here, a generic but innovative blockchain-IoT architecture is proposed to improve the scalability and interoperability of IoT.

In the healthcare sector, a blockchain-based solution was used to enhance patient care. The approach was based in the use of smart contracts, deployed on the Ethereum blockchain platform, to store patient data securely and to enable a transparent communication between healthcare individuals and patients [22]. Reliable data communication in IoT networks [23] has been addressed using a three-layer methodology based on LoRa, blockchain, and Ethereum Swarm technologies. Blockchain has been used with a novel distributed ledger technology for digital monitoring of environmental ecosystems [24].

Blockchain-based solutions have been also used to improve privacy and security in IoT devices [32], [33]. Here, a Blockchain Connected Gateway was used with a Bluetooth Low Energy (BLE) device to prevent unauthorised access to sensitive data. The challenge of resource and memory constrained in IoT devices has been also addressed by integrating mobile blockchain-technology with edge computing [34]. Here, edge computing brings an efficient

TABLE 1. Comparison between Blockchain-IoT architectures.

Ref.	Authentication	Scalability	Access Control	Data Integrity	Interoperabillity	Privacy	Decentralised	BC Type/ Consensus
[25]	✓	✗	✓	✓	✓	✓	✓	Public/PoW
[26]	✓	✗	✓	✗	✗	✓	✓	NA
[27]	✓	✓	✓	✓	✓	✓	✓	Public
[28]	✓	✗	✓	✓	✓	✓	Partial	Private/PoW
[16]	✓	✗	✓	✓	✗	✓	✓	Hybrid/Nil
[29]	✓	✗	✗	✓	✓	✓	✓	Private/Nil
[23]	✓	✗	✓	✓	✗	✗	✓	Private/Nil
[30]	✓	✓	✓	✓	✓	✓	✓	Public/PoW
[31]	✓	✗	✓	✗	✗	✗	✓	NA
[21]	✓	✓	✓	✓	✓	✓	✓	NA

mechanism to achieve consensus between the mobile device and the blockchain network such that the scalability and efficiency are highly improved. The challenge of trust and secure storage of massive data from IoT devices was addressed by [35]. In this approach, a distributed-blockchain data storage scheme was introduced to efficiently store the data using certificateless cryptography. A Directed Acyclic Graph (DAG)-based blockchain is proposed to secure and store sensitive data in industrial IoT (IIoT) applications [36].

Hacking vulnerabilities and privacy concerns in IoT devices have been also addressed with blockchain-based solutions [26], [28], [37]. For the hacking issue, a colour spectrum chain blockchain technique [38] was used with the Thin Plate Spline (TPS) to assess different security levels and enhance IoT devices' security. In [27] a blockchain architecture was proposed for decentralized authentication [29], [30], [31] in resource-constrained IoT devices. Similarly, in [25] a Fog computing-based decentralized authentication model for lightweight IoT devices was proposed. The key idea of this approach, is to permit communication between devices from different platforms by means of a distributed authentication and authorisation blockchain mechanism. Table 1 summarizes the main features of each blockchain-IoT architecture discussed in this section.

A. CONTRIBUTIONS AND PAPER OUTLINE

In this paper, we aim to contribute in the research of blockchain-IoT to improve trust and security. The research is inspired in a decentralized blockchain architecture that addresses the aforementioned vulnerabilities and security requirements of IoT networks. To this end, smart contracts and distributed ledger technology are used in the design of authentication, authorisation and access control algorithms. The combined contribution gives a coherent, secure, and reliable blockchain-IoT environment that can be seamless integrated in IoT networks.

The outline of the paper is as follows. Section III gives the proposed blockchain-IoT methodology. Section IV describes the elements of the proposed architecture. Section V reports the results in a real-world IoT environment. Conclusions and future work are shown in Section VI.

III. METHODOLOGY

The proposed methodology is inspired in previous works [21], [25], [26], [27], [28], [29], [30], [37] in blockchain-IoT that cover the aforementioned security requirements in IoT environments. The model proposed in this work exploits the use of smart contracts in the distributed architecture of blockchains to provide useful tools for authentication and access control. These tools manage the interaction between IoT devices and users in a decentralized IoT environment.

Fig. 2 depicts the high-level architecture of the proposed model. Here, smart contracts are used in blockchain-technology to enforce the authentication and access control rules for IoT devices and users. This, in consequence, will allow seamless communication [39] and secure data sharing between the devices and users registered in an heterogeneous IoT environment. The following elements can be identified from Fig. 2:

- **IoT Administrator (Admin)**: is responsible for designing and deploying the smart contracts, and to enforce the mechanisms for interaction between IoT devices and users. Once the smart contract is deployed, the administrator initializes the network where all the IoT users, managers and the respective devices are registered in the blockchain network using their accounts and credentials.
- **IoT Managers**: are responsible to allocate the resources of a specific IoT zone by following the established IoT protocols. The administrator registers each IoT managers individually using their respective public addresses and, in turn, they are provided with a system identification (SID).
- **IoT devices**: are represented by a diverse class of sensors, actuators and end-devices that cannot participate in blockchain transactions by their own. These devices are registered and connected to their respective IoT manager using standard IoT architectures and protocols. The IoT devices are registered in the blockchain network using the SID of the respective IoT manager and then they are mapped into the network. An authentication key is given to the IoT manager for each successfully registered device.

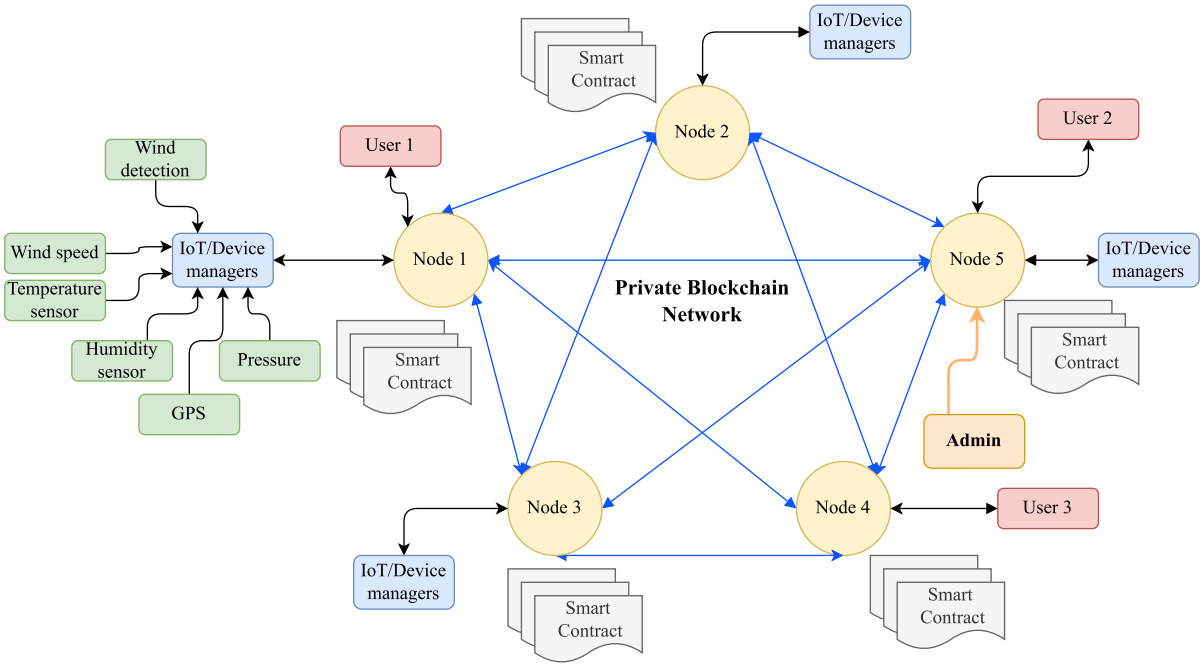


FIGURE 2. High-level architecture of the proposed model.

- **IoT users**: are users that are registered by the administrator to obtain access to the IoT resources. To this end, an unique user ID (UID) and authentication keys are provided.
- **Blockchain network**: in this paper, a private blockchain network is used due to its scalability, low latency, and privacy. Furthermore, private blockchains use consensus mechanisms to achieve fast transaction processing in large-scale IoT networks. Each transaction executed by the IoT managers and users is verified using the consensus mechanisms and mined into a new blockchain block. All blockchain nodes communicate with each other to synchronize the devices' encrypted data and the data associated with the authentication, authorization, and access control in the distributed ledger.
- **Smart Contracts**: enforce the authentication and access control rules for IoT devices and users. These contracts are applied on Ethereum Virtual Machines (EVMs) in each node of the blockchain to enforce registration, authentication, and access control rules for device-to-device and device-to-user interactions. Furthermore, the smart contract is responsible to regulate the registration procedure (for the administrator), authentication of devices and users, and enable the access control for accessing and uploading IoT data.

A. BLOCKCHAIN TYPE AND CONSENSUS MECHANISM

Consensus mechanisms are applied in the network nodes of Fig. 2 to validate the transactions involving blockchain networks. Here, the type of blockchain used in the architecture and the consensus mechanism play a major role in the scalability, latency, energy efficiency, performance, and security of the overall IoT environment.

Proof of Work (PoW) is used as a consensus mechanisms in public blockchain platforms such as Bitcoin and Ethereum. However, these methods demand high computational power and require a large amount of time to find a good solution [40]. These disadvantages can degrade the performance of the blockchain-IoT in a real-world implementation.

Proof of Stake (PoS) and Practical Byzantine Fault Tolerance (PBFT) are other class of consensus mechanisms that improve the throughput, transaction time and energy efficiency in contrast to PoW [41], [42]. Since, the proposed architecture is intended to be applied in a real-world IoT setting, then a private blockchain with PoS or PBFT as consensus mechanisms is used to increase the security and trust of the overall IoT network. The advantages of using private blockchain is that the scalability is enhanced, with higher transaction throughput, fast consensus and better energy efficiency. These benefits make the private blockchain ideal for real-time IoT environments with limited resources and with fast secure transactions requirements across different IoT zones.

B. ENCRYPTION OF DATA AND TRANSACTIONS

The transactions executed in the blockchain network are digitally signed. In addition, the ECDSA algorithm is used to generate a pair of public/private keys for the encryption/decryption of each transaction started by either IoT administrator, managers or users [43].

The encryption process is as follows: i) a member of the IoT network (admin, manager, user) starts a transaction on

blockchain and the hash value of the transaction data is digitally signed by its unique private key, ii) after signed, then it is broadcaster to the blockchain network containing the sender's public key, recipient's address, transaction data, and the member digital signature, iii) this transaction is received and validated by either validators or miners, and iv) when the validation is successful, the transaction is incorporated in a new block and added to the blockchain's ledger by means of the consensus mechanism. The complete encryption process defines a reliable and secure methodology to guarantee that the integrity of the data is not compromised.

C. INTERACTIONS IN THE IOT NETWORK

Each member of the IoT network has well-defined activities when they interact between each other. These interactions are:

- **Administrator-to-IoT Manager and Users**: the Administrator registers both the users, IoT managers and the devices to their corresponding IoT zone.
- **Administrator-to-Blockchain**: the Administrator defines and deploy the smart contract over the blockchain and no changes can be made once deployed.
- **IoT Manager-to-Blockchain**: a web3 provider is used to connect each IoT manager to a blockchain node. A JSON RPC protocol is used for the connection. The blockchain network authenticates, via smart contract rules, the IoT manager and its associated devices once they are registered. Therefore, data upload and stored is granted and is immutable over the network.
- **Device-to-IoT Manager**: here, the end-devices are connected and managed by their respective IoT managers under traditional IoT security protocols. Therefore, the IoT managers are responsible for the registration of their respective end-devices over the blockchain.
- **Users-to-Blockchain**: a web3 provider is used to connect each user to a blockchain node. A JSON RPC protocol is used for the connection. The blockchain network authenticates, via smart contract rules, the user once it is registered. Therefore, access to uploaded data is granted depending the access control list established by the Administrator in the smart contract.

D. AUTHENTICATION AND ACCESS CONTROL

Each member of the IoT network requires to register to blockchain using the services of web3 providers, e.g., Infura, Alchemy and MetaMask. In addition, a set of authentication keys, saved in the smart contracts, are assigned to each connected user and device in the blockchain network. These keys are verified by the blockchain network in order to give access to the network each time a transaction is requested. A consensus mechanism is then applied by blockchain nodes to verify the transactions before mining them into new blocks.

The data of the authenticated devices is uploaded, as data strings, over the blockchain network by means of well-defined functions in the smart contract. Here, the administrator establishes an access control list that regulates the request of data retrieval of uploaded data. A data request created by a user is granted after the access control in the smart contract is verified. This process is applied every time a transaction is requested by either the IoT managers or users.

IV. ELEMENTS OF THE PROPOSED ARCHITECTURE

The communication between the members of the Fig. 2 is divided into phases: initialization phase, authentication phase, and data exchange phase. Each of these phases are implemented in diverse zones of the IoT environment. Each zone is managed by its respective IoT manager under standard IoT protocols.

A. INITIALIZATION PHASE

This phase is dedicated to register each member of de IoT network, i.e., IoT managers, users, and devices in the blockchain network. This is done by following the smart contracts' rules established by the blockchain administrator. Here, various functions are written to register each stakeholder in the blockchain. These functions can only be called by the blockchain administrator. This phase has four main algorithms dedicated to the registration of either the IoT manager, device, or user, and the design of the access control list. Fig. 3 depicts each element of the initialization phase.

1) IOT MANAGER REGISTRATION

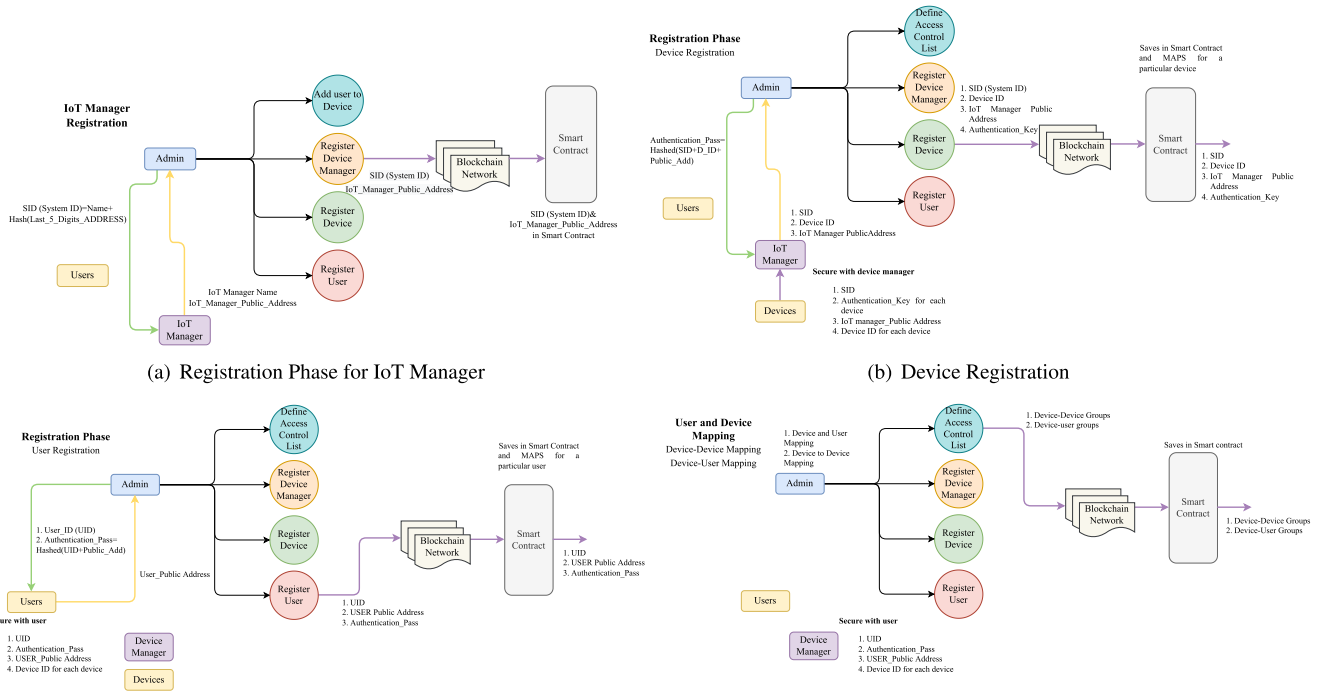
Fig. 3(a) shows the IoT manager registration scheme. Here, the blockchain administrator is responsible of the registration of the IoT managers that are in charge of the IoT zones. This is done by creating a mapping for each IoT manager using a unique system ID (SID). After the IoT manager is registered, then a block is mined and broadcasted to the nodes of the blockchain. The process of the IoT manager registration is stated as follows:

1) The IoT manager requests a blockchain registration to the administrator given its name and public address (Public_Add).
2) The administrator uses any web3 provider to initialize a transaction for the IoT manager registration in the smart contract. This transaction, denoted as T_1, is realized using the following function in the smart contract

$$T1 = \text{Register\_IoT\_Manager(Name, Public\_Add)}. \tag{1}$$

3) The blockchain nodes and smart contract analyses the transaction by checking if the name of the IoT manager and its public address have been previously registered. After the transaction is approved and registered, a unique SID is given by the smart contract which is mapped to IoT manager's name and public address in the distributed ledger of the blockchain as follows:

$$\begin{aligned} \text{SID} = {} & \text{Name@Last 5 digits of Public\_Add} \\ & \text{IoT Manager} \rightarrow \{\text{IoT Manager Name, SID,} \\ & \text{Public\_Add}\} \end{aligned} \tag{2}$$



(a) Registration Phase for IoT Manager

(b) Device Registration

(c) User Registration

(d) Defining Access Control List

FIGURE 3. **Initialization phase elements.**

4) The administrator can retrieve and distribute the generated SID using a call function of the blockchain.

2) DEVICE REGISTRATION

Fig. 3(b) shows the device registration scheme. All the devices connected to the registered IoT managers are registered using the respective IoT manager SID. An authentication key (Auth_Key) is generated by a Keccak-256 hashing algorithm and returned to the IoT manager for each successfully registered device. This key will be used to authenticate the device for data uploading. Finally, the registered devices and their respective Auth_Keys are transferred to the respective IoT managers in the distributed ledger and then a block is mined and distributed to blockchain nodes. The process of device registration is stated as follows:

1) The IoT manager requests a device registration to the administrator given the device ID (DID), SID, and the IoT manager's public address.
2) The administrator uses any web3 provider to initialize a transaction for the device registration. This transaction, denoted as T_2, is realized using the following function in the smart contract

$$T_2 = \text{Register\_Device(DID, SID, Public address)}. \tag{3}$$

3) The blockchain nodes and smart contracts authenticate the transaction by checking the IoT manager's SID and public address, and if a valid mapping exists in the smart contract. In addition, the uniqueness of the DID is analysed. If successful, the device is registered and mapped to its respective IoT manager. Then, an Auth_Key is provided by the smart contract and mapped to the DID, IoT manager's SID, and Public_Add in the distributed ledger of the blockchain as follows:

$$\begin{aligned} &\text{Auth\_key} \\ &\quad = \text{Keccak-256(DID, SID, Public\_Add)} \\ &\quad \text{Device} \rightarrow \{\text{DID, SID, Public\_Add, Auth\_Key}\} \end{aligned} \tag{4}$$

4) The administrator can retrieve and distribute the generated Auth_Key using a call function of the blockchain.

3) USER REGISTRATION

The users registration follows a similar procedure to the IoT managers and devices registration. The administrator registers the user using the established rules in the smart contract. Here, both an Auth_Key and user ID (UID) are assigned for authentication data access in the blockchain. These credentials are stored and mapped in the blockchain network, then a block is mined and broadcaster to the nodes of the blockchain. The process of the user registration is stated as follows:

1) The user requests a user registration to the administrator given its name and Public_Add.
2) The administrator uses any web3 provider to initialize a transaction for the user registration in the smart

contract. This transaction, denoted as T_3, is realized using the following function in the smart contract

$$T_3 = \text{Register\_User(Name, Public\_Add)}. \quad (5)$$

3) The blockchain nodes and smart contracts authenticate the transaction by analysing the uniqueness of the name and Public_Add of the user. If successful, the user is registered in the blockchain and a unique UID and Auth_Key are assigned and mapped to the name and Public_Add of the user in the distributed ledger of the blockchain as follows:

$$\begin{aligned} &\text{UID} \\ &= \text{Name@Last 5 digits of Public\_Add} \\ &\text{Auth\_key} = \text{Keccak-256(UID, Public\_Add)} \\ &\text{User} \rightarrow \{\text{Name, UID, Public\_Add, Auth\_key}\}. \end{aligned} \quad (6)$$

4) The administrator can retrieve and distribute the generated UID and Auth_key using a call function of the blockchain.

4) ACCESS CONTROL LIST

An access control list needs to be established in the smart contract after the IoT managers, users, and devices were successfully registered. This enables the regulation of the user's access to the uploaded IoT data. This implies that the administrator needs to define a device-to-user mapping, based on the access control list, that constraints the accessibility of users to data uploaded by a specific device. The process for the access control list design is stated as follows:

1) The administrator uses any web3 provider to initialize a transaction for the addition of a new user or device mapping. The transaction, denoted as T_4, uses the following function of the smart contract

$$T_4 = \text{Map\_in\_Access\_Control\_list(UID, DID)}. \quad (7)$$

2) Blockchain checks the uniqueness of the UID and DID and the associated mapping in the access control list to validate the transaction. If the mapping does not exist in the access control list, then a new control mapping for the UID and DID is created and added to the list. This allows the user to call functions in the smart contract and get access to the data uploaded by a device. This is done by using the following function

$$\text{Access\_Control\_List} \rightarrow \{\text{UID, SID}\}. \quad (8)$$

B. AUTHENTICATION AND DATA EXCHANGE PHASES

Fig. 4 shows the authentication and data exchange schemes of the proposed model. For the authentication and data upload case, the registered users and devices, under the access control list, are authenticated by the blockchain nodes. Then, they can start transactions in the blockchain network

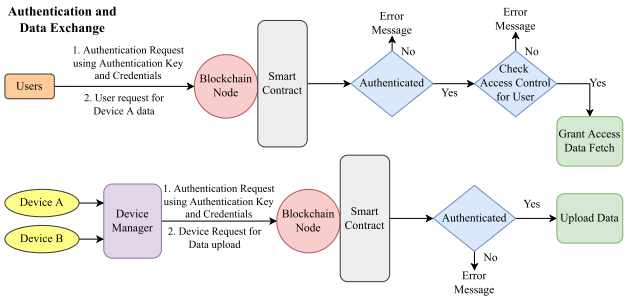


FIGURE 4. Authentication and exchange of data schemes.

to either upload or access data using any web3 provider. For authentication and data access, a valid and authenticated user can mine the data uploaded by a device using the call functions available at the smart contract.

The authentication and data upload process is as follows:

1) The IoT managers, in charge of their respective users and devices, use web3 providers to start a blockchain transaction for device data upload. Here, it is provided the respective public addresses, the device's Auth_key, DID and SID, and the data string to be uploaded. This transaction, denoted as T_5, is defined by the following function in the smart contract

$$\begin{aligned} T_5 = \text{Upload\_Data(DID, SID, Public\_Add,} \\ \text{Auth\_key, Data\_String)}. \end{aligned} \quad (9)$$

2) The blockchain validates and authenticates the transaction each credential of the device and IoT manager.
3) A new block is mined and distributed to the blockchain nodes which includes the DID and the uploaded data string as

$$\text{Device} \rightarrow (\text{DID, Data\_String}). \quad (10)$$

4) The users with device access (according to the access control list) can retrieve the uploaded data using the call functions.

The authentication and data access process is as follows:

1) A registered user starts a transaction in the blockchain for device data access using any web3 provider by giving its UID, Auth_Key, Public_Add, and the device's DID. The transaction, denoted as T_6, is given by the following function

$$T_6 = \text{Data\_Access(UID, Auth\_key, Pub\_Add, DID)}. \quad (11)$$

2) The transaction is verified and authenticated. I addition, the blockchain verifies if the access control list contains a mapping for UID and DID.
3) If the authentication is successful, then the blockchain returns the device data string as

$$\text{Device data} \rightarrow \text{Data\_String}, \quad (12)$$

that can be accessed by the registered user using its user's Public_Add.

V. RESULTS

The proposed model is verified in an IoT simulated environment to evaluate its safety and reliability capabilities in a decentralized IoT environment. Fig. 5 depicts the implementation diagram of the proposed model.

The members of the IoT network, i.e., administrator, IoT managers, users, and devices, are registered to the blockchain network using any web3 provider. Each IoT member has a blockchain account with private and public keys generated by MetaMask. These keys are used to sign the blockchain transactions. Members of the IoT use the JSON RPC protocol to connect with the web3 service and facilitate the interaction with the blockchain nodes. The proposed architecture is first tested and optimised using the Ganache blockchain simulator. When the final model is obtained, then it is deployed in the Polygon Testnet platform.

Once the smart contract is deployed and the administrator starts the network, then all members of the IoT network can request for the different available functions written in the smart contract. These functions cover each element of the initialization, authentication and data exchange phases.

The smart contract is developed using the remix IDE of Solidity. This web platform allows to write, test, compile, debug, and deploy EVM-compatible smart contracts on the Ethereum blockchain without using diverse frameworks. The application binary interface (ABI) and Bycode are used to compile the contract before its deployment in the blockchain network (either Ganache and Polygon Testnet). The overall process is observed in Fig. 6.

The designed and deployed contract covers six main functions (see Fig. 7) with additional inherent functions of the smart contract. The smart contract is rigorously tested to fulfil the IoT security requirements. Then, the injected web3 service provider is used to connect the MetaMask wallet with the imported accounts of the blockchain simulation networks Ganache and Polygon.

A. SMART CONTRACT TESTING ON GANACHE

1) INITIALIZATION PHASE

An access control list is defined by the Ethereum account of Ganache to register IoT managers, users, and the corresponding devices. MetaMask account is used to digitally sign each input data of every registration and access control transaction request. After confirmation of the MetaMask transaction request by the administrator, digitally signed transactions were executed and then mined as a new block on the blockchain distributed ledger (see Fig. 8).

2) DEFINING ACCESS CONTROL LIST

An access control list is defined using the administrator Ethereum account after each IoT member is registered. Here, the administrator initializes each transaction to map the corresponding DIDs to UIDs. Then, digitally signed transactions are executed followed by mining as a new block in the distributed ledger (see Fig. 9).

3) AUTHENTICATION AND DATA UPLOAD

The registered IoT managers use their respective MetaMask accounts to initialize transactions for data upload of devices using the SID, DID, and authentication key. Once the transactions are confirmed by the MetaMask user account, then the transactions are digitally signed, credentials are authenticated by the blockchain nodes and then mined as a new block into the distributed ledger. Fig. 10 shows an example of data upload transaction of the string "*Latitude 49.7749° N, Longitude: 122.4194° W*".

4) DATA RETRIEVAL: AUTHENTICATION AND ACCESS CONTROL

The users registered in the blockchain using their respective MetaMask accounts can fetch the uploaded data by the registered devices using the fetch data call function. Here, the access to read the data is granted only if permitted by the access control list. Fig. 11 shows an example of data retrieval of the following string "*Latitude: 49.7749° N, Longitude: 122.4194° W*".

To evaluate the performance of the architecture, the following experiments are conducted: i) data upload transactions and ii) data fetch of both the IoT manager and users.

B. TRANSACTION COMPLETION AND BLOCK VALIDATION TIME

Random transactions are simulated to determine both the transaction completion (TCT) and block validation (BVT) times. Fig. 12 shows the obtained results. We can observe that the average time to complete a transaction is approximately 4.18 sec, whilst the block validation is around 2.14 sec. We further compute the completion time for 100 data upload transactions. The results are given in Fig. 13 where consistent results are obtained in comparison with Fig. 12. In this case, the average time of a transaction completion is approximately 4.25 sec and 2.22 sec for the block mining. The observed variations in the block validation time can be associated to the block size, network congestion and latency, the number of transactions in a block or the computational complexity of the functions in the smart contract. Notice that the average time for block validation is approximately 2.22 sec which is lower than other blockchain networks with different consensus mechanisms, e.g., Bitcoin requires approximately 10 min for PoW and up to 19 sec for PoW in Ethereum. The variations in the transaction time can also be attributed to the congestion and latency of the network which can be defined as

$$\mathrm{TCT} = \mathrm{BVT} + \mathrm{NLT}, \tag{13}$$

where NLT is the network latency time. We additionally compute the transactional throughput time for data transactions realized by IoT managers. Here, the average throughput for data upload transactions is approximately 9.69 transactions per minute. This time can be further improved by using private blockchain networks with faster consensus mechanisms. These are coherent requirements in

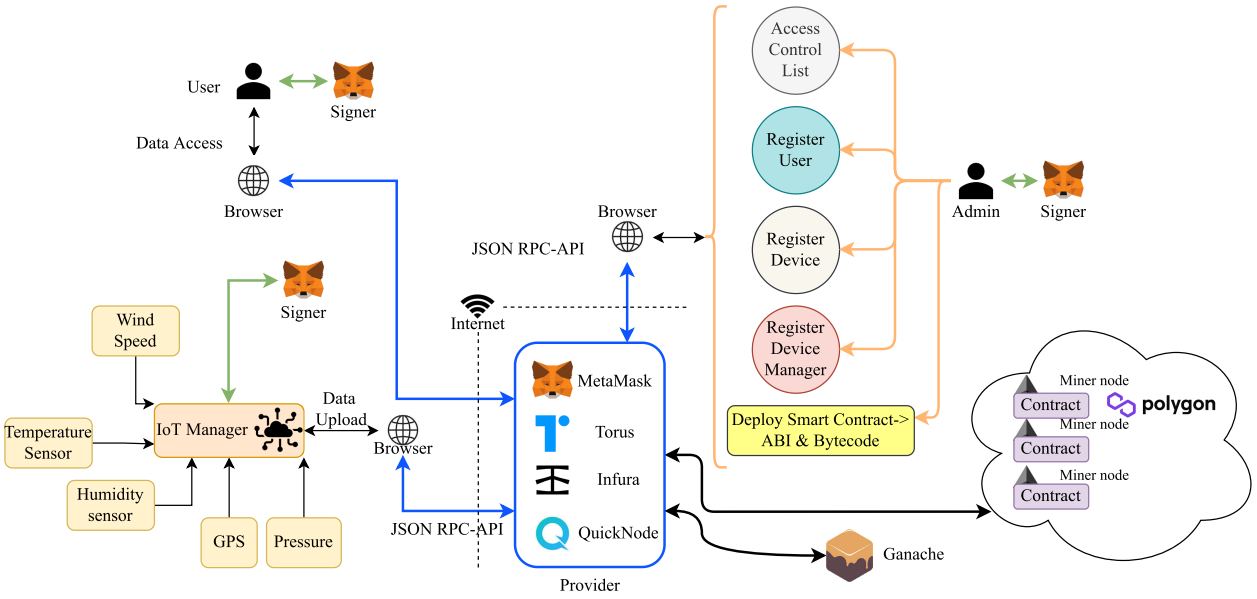


FIGURE 5. Implemented model design.

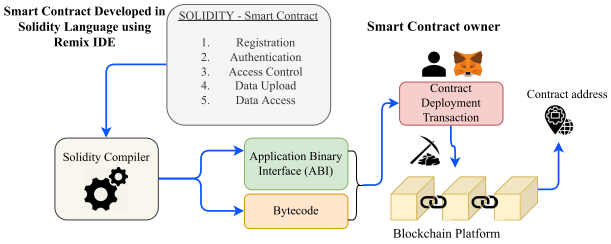


FIGURE 6. Smart contract deployment.

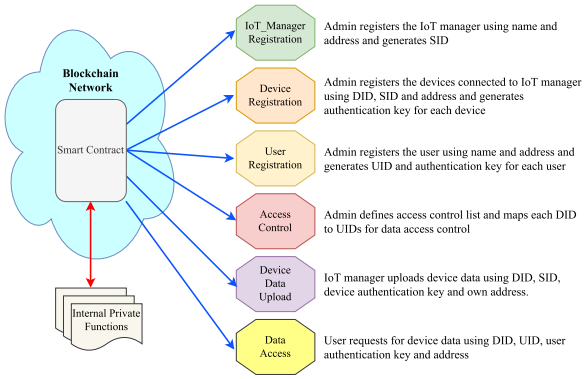


FIGURE 7. Smart contract breakdown.

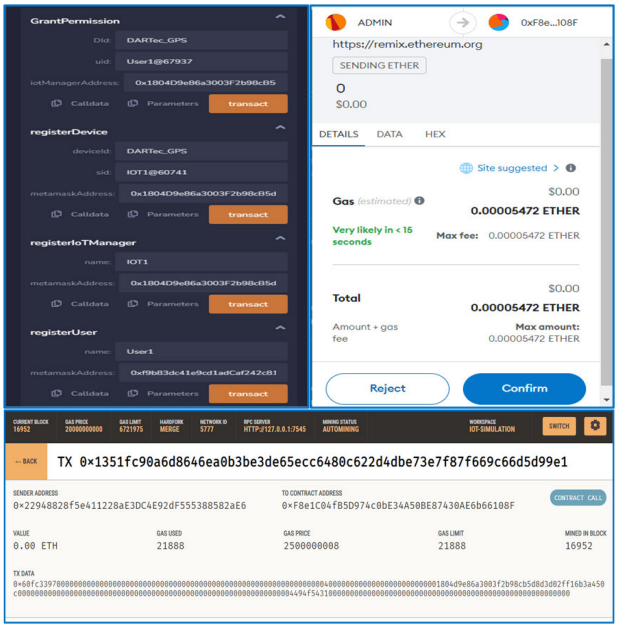


FIGURE 8. IoT manager registration over Ganache.

current real-time data upload and exchange demand in IoT networks.

C. GAS AND CPU CONSUMPTION

The gas consumption is used to measure the computational effort of the proposed approach. This is an important factor that can imply money loss in IoT applications. The gas consumption depends mainly on the complexity and resource requirements for each transaction. Here, it is evident that the gas consumption is increased when the transactions are complex and demand more computational resources. Fig. 14 shows the gas consumption for data upload transactions in the Polygon Testnet. The results show consistent gas consumption across all the transactions.

Additionally, Fig. 15 shows the CPU consumption which further demonstrate the benefits of the proposed model.

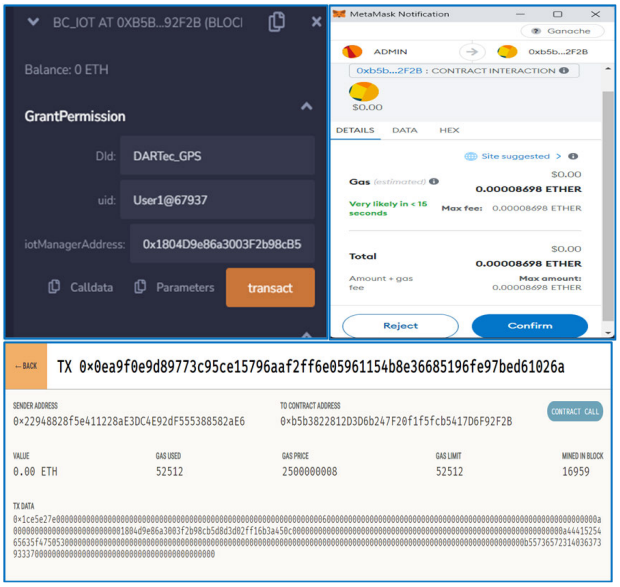


FIGURE 9. Defining access control list.

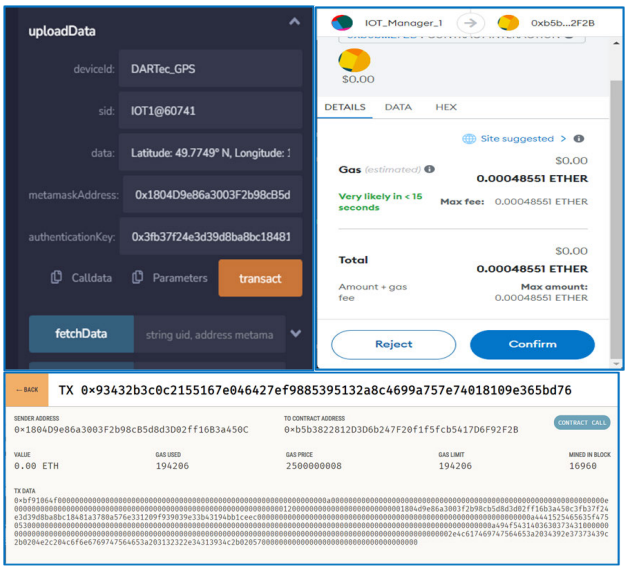


FIGURE 10. Data upload transaction.

D. DATA RETRIEVAL ANALYSIS

Fig. 16 shows the real-time retrieval of the device data uploaded by the IoT manager. The results show an average retrieval time of approximately 251.44 msec. Here the data retrieval call functions in the smart contract play the most important role in the data retrieval time, which can also be affected by the latency and congestion of the network.

E. PERFORMANCE IN CONGESTED NETWORKS

The proposed model is also assessed under a congested networks. Figs. 17-19 show the transaction time, the gas and CPU consumption results in a congested network. The results are consistent with the previous ones, where the main difference appears in well-identified peaks of transaction time caused by the congested network. Nevertheless, the results show low gas and CPU consumption and a relatively fast transaction time.

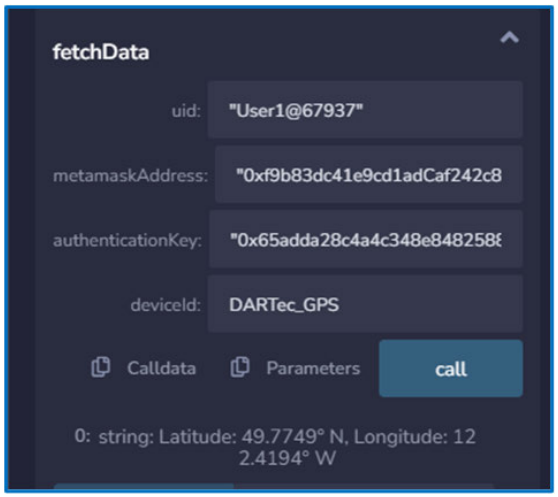


FIGURE 11. Data retrieval by registered users as per access control list.

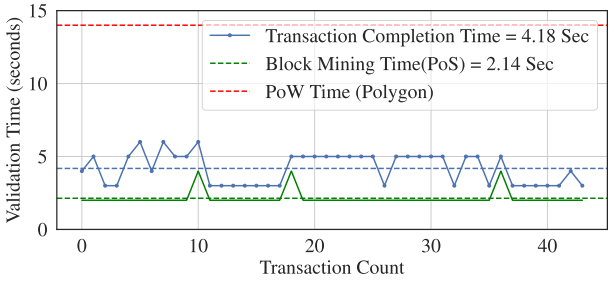


FIGURE 12. Transaction completion and block validation times using random transactions.

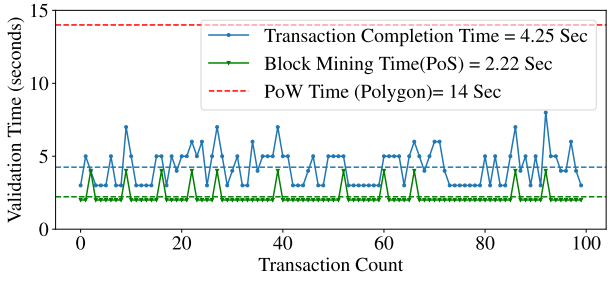


FIGURE 13. Transaction completion and block validation times for 100 data upload transactions.

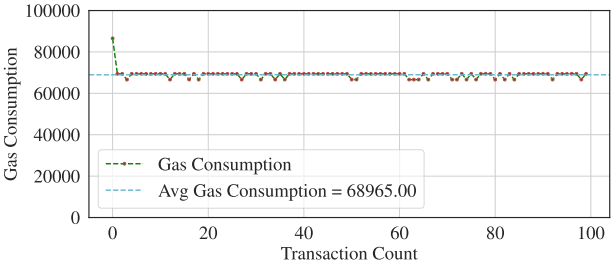


FIGURE 14. Gas consumption analysis.

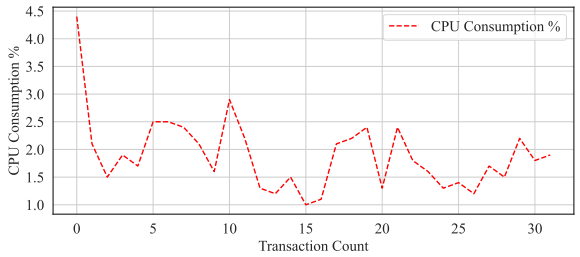


FIGURE 15. CPU consumption analysis.

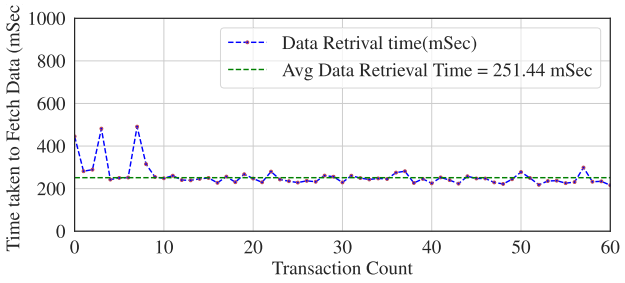


FIGURE 16. Data retrieval analysis.

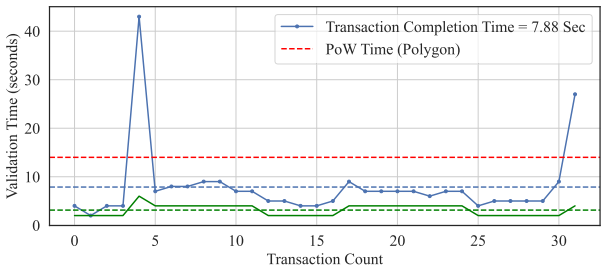


FIGURE 17. Transaction completion on a congested network.

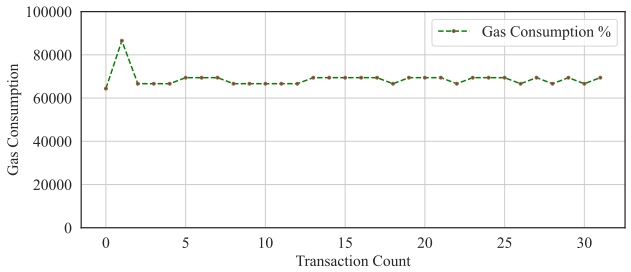


FIGURE 18. Gas consumption on a congested network.

F. EVALUATION AGAINST SECURITY ATTACKS

The proposed model exhibits robust properties against a diverse vulnerabilities and possible attacks in the IoT environment. Here, the use of smart contracts highly improve the authentication, access control, and data exchange rules, which are crucial elements to guarantee security in the overall IoT system. The main features observed in the simulations are: i) the unauthorized access is improved by limiting the access to the IoT resources to only authenticated and validated users and devices. ii) Data tampering is prevented

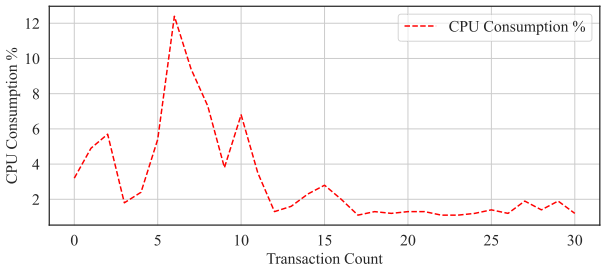


FIGURE 19. CPU consumption on a congested network.

by using the distributed ledger and digital signatures. Here, only authorized IoT managers are granted to upload device data and, once uploaded, the data becomes part of the immutable ledger such that any attempt to modify or alter the data will be rejected by the network. iii) The proposed model is protected against replay attacks since each transaction is verified and recorded by the IoT network. Each transaction is associated to an unique identifier hash and a time stamp. In consequence, the transactional data stored in the digital ledger cannot be duplicated or transmitted multiple times within the network. iv) The proposed model uses digital signatures, hashing algorithms and encryption techniques to ensure security against Man-in-the-Middle (MITM) attacks. The communication between nodes are digitally signed such that they can only be decrypted by authorized users. This prevents attackers from intercepting and modifying data during transmissions. v) Robust decentralized authentication mechanisms are used by the proposed model to ensure the participation of only authorized users with their respective unique identities. Here, the use of authentication keys and mappings to public addresses provide an additional security layer to the IoT environment against impersonation attacks. vi) The proposed model is robust against spoofing attacks. Here, a successful spoof identity attack requires the DID, SID, and the associated device private key. The authentication mechanisms used in the proposed model ensures that only authorized devices with their unique private keys can interact with the IoT system. vii) Robust Identity and authentication mechanisms are used by the proposed model to assign unique IDs and verifiable identity based on the public address registered in the network. This helps to avoid the creation of multiple fake identities to manipulate the system. Furthermore, transactional data are signed with the respective IoT manager private keys to reduce the probability of generating fake identities. viii) The decentralized nature of the blockchain can mitigate the distributed denial of service (DDoS) attacks. Here, the smart contracts and data are distributed across multiple nodes which implies that there are no single point failures and hence, the IoT system is highly resilient. ix) Data transactions cannot be repudiated due to the use of digital signatures in the smart contract.Here, a recorded transaction cannot be denied by any member of the blockchain network. This enhances the accountability and

transparency of the IoT system. x) Sensitivity data remain secure by enhancing the access control rules. This reduces the risk of data exposure and privacy breaches.

G. DISCUSSIONS

The results of the implementation of the proposed model on Polygon Testnet as a proof of concept demonstrate the effectiveness of smart contracts as solution to enforce authentication, access control, and data exchange. However, some challenges can be identified such as: 1) the use of web3 providers can add delays in the transaction time attributed to the network latency, transactional overload and network congestion, which can hinder the adoption of the proposed model in some IoT scenarios that require real-time data exchange. The implementation on private blockchain networks that are built in platforms such as Hyperledger is highly recommended since they are based on adaptable consensus mechanisms (e.g., Practical Byzantine Fault Tolerance (PBFT)) which notably decreases the transaction time and, therefore, the interoperability and scalability. Furthermore, the absence of gas consumption on private blockchains eliminate transaction costs such that the system is more cost-efficient. 2) IoT scenarios are prone to generate large amount of data which can lead to inefficiency and scalability issues on the blockchain-technology. This problem can be alleviated by incorporating distributed storage solutions which provide a decentralized and distributed storage network that can effectively store the data maintaining its integrity and security.

VI. CONCLUSION

This paper proposes a novel blockchain-IoT models based on smart contracts and distributed ledger to improve key security requirements in IoT environments: authentication, access control, and data exchange. The integration of smart contracts and distributed ledger in the IoT network increases the security and trustworthiness of IoT environments by enhancing the mechanisms for authentication, access control and data exchange. The proposed model is implemented in Polygon Testnet blockchain simulator and applied in a simulated IoT environment. The simulations verify the effectiveness of the approach in terms of the transaction throughput, block mining time, and transaction completion time, whilst satisfying the IoT security requirements. In the future, blockchain technology will see significant advancements in managing the vast amount of data generated by the Internet of Things (IoT). This will be achieved through the integration of distributed storage solutions like IPFS (InterPlanetary File System) and SWARM, which provide secure and decentralized storage for IoT data. Additionally, transaction speed and throughput on blockchain networks will be improved by adopting alternative platforms such as Hyperledger, which uses the efficient PBFT (Practical Byzantine Fault Tolerance) consensus mechanism. These developments are vital for ensuring the integrity and scalability of IoT data management on blockchain networks.

REFERENCES

[1] S. Al-Rubaye, J. Rodriguez, L. Z. Fragonara, P. Theron, and A. Tsourdos, "Unleash narrowband technologies for industrial Internet of Things services," *IEEE Netw.*, vol. 33, no. 4, pp. 16–22, Jul. 2019.

[2] Y. He, Y. Ren, Z. Zhou, S. Mumtaz, S. Al-Rubaye, A. Tsourdos, and O. A. Dobre, "Two-timescale resource allocation for automated networks in IIoT," *IEEE Trans. Wireless Commun.*, vol. 21, no. 10, pp. 7881–7896, Oct. 2022.

[3] H. Whitworth, S. Al-Rubaye, A. Tsourdos, J. Jiggins, N. Silverthorn, and I. Khan, "An information entropy and ensemble learning approach for DR-DOS detection within aviation networks," in *Proc. Integr. Commun., Navigat. Surveill. Conf. (ICNS)*, Apr. 2022, pp. 1–12.

[4] W. Guo, Z. Wei, O. Gonzalez, A. Perrusquía, and A. Tsourdos, "Control layer security: A new security paradigm for cooperative autonomous systems," *IEEE Veh. Technol. Mag.*, early access, pp. 2–11, 2023, doi: 10.1109/MVT.2023.3290773.

[5] R. Kumar and R. Sharma, "Leveraging blockchain for ensuring trust in IoT: A survey," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 10, pp. 8599–8622, Nov. 2022.

[6] A. Aloseel, S. Al-Rubaye, A. Zolotas, and C. Shaw, "Attack-detection architectural framework based on anomalous patterns of system performance and resource utilization—Part II," *IEEE Access*, vol. 9, pp. 87611–87629, 2021.

[7] B. Fraser, S. Al-Rubaye, S. Aslam, and A. Tsourdos, "Enhancing the security of unmanned aerial systems using digital-twin technology and intrusion detection," in *Proc. IEEE/AIAA 40th Digit. Avionics Syst. Conf. (DASC)*, Oct. 2021, pp. 1–10.

[8] S. Luthra, D. Garg, S. K. Mangla, and Y. P. S. Berwal, "Analyzing challenges to Internet of Things (IoT) adoption and diffusion: An Indian context," *Proc. Comput. Sci.*, vol. 125, pp. 733–739, Jan. 2018.

[9] A. Gupta, R. Christie, and P. R. Manjula, "Scalability in Internet of Things: Features, techniques and research challenges," *Int. J. Comput. Intell. Res.*, vol. 13, no. 7, pp. 1617–1627, 2017.

[10] D. Jiang, F. Wang, Z. Lv, S. Mumtaz, S. Al-Rubaye, A. Tsourdos, and O. Dobre, "QoE-aware efficient content distribution scheme for satellite-terrestrial networks," *IEEE Trans. Mobile Comput.*, vol. 22, no. 1, pp. 443–458, Jan. 2023.

[11] V. Ahlqvist, P. Holmberg, and T. Tangerås, "A survey comparing centralized and decentralized electricity markets," *Energy Strategy Rev.*, vol. 40, Mar. 2022, Art. no. 100812.

[12] V. K. N. Lau, S. Cai, and M. Yu, "Decentralized state-driven multiple access and information fusion of mission-critical IoT sensors for 5G wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 869–884, May 2020.

[13] S. Zeba, P. Suman, and K. Tyagi, "Types of blockchain," in *Distributed Computing to Blockchain*. Amsterdam, The Netherlands: Elsevier, 2023, pp. 55–68.

[14] H. F. Atlam, M. A. Azad, A. G. Alzahrani, and G. Wills, "A review of blockchain in Internet of Things and ai," *Big Data Cognit. Comput.*, vol. 4, no. 4, p. 28, 2020.

[15] P. Patil, M. Sangeetha, and V. Bhaskar, "Blockchain for IoT access control, security and privacy: A review," *Wireless Pers. Commun.*, vol. 117, no. 3, pp. 1815–1834, Apr. 2021.

[16] T. Rathee and P. Singh, "A systematic literature mapping on secure identity management using blockchain technology," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5782–5796, Sep. 2022.

[17] S. Jain, "Can blockchain accelerate Internet of Things (IoT) adoption," Deloitte, Zürich, Switzerland, Tech. Rep., 2021.

[18] V. A. Siris, D. Dimopoulos, N. Fotiou, S. Voulgaris, and G. C. Polyzos, "OAuth 2.0 meets blockchain for authorization in constrained IoT environments," in *Proc. IEEE 5th World Forum Internet Things (WF-IoT)*, Apr. 2019, pp. 364–367.

[19] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Future Gener. Comput. Syst.*, vol. 82, pp. 395–411, May 2018.

[20] J. Sengupta, S. Ruj, and S. D. Bit, "A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT," *J. Netw. Comput. Appl.*, vol. 149, Jan. 2020, Art. no. 102481.

[21] D. Pavithran, K. Shaalan, J. N. Al-Karaki, and A. Gawanmeh, "Towards building a blockchain framework for IoT," *Cluster Comput.*, vol. 23, no. 3, pp. 2089–2103, Sep. 2020.

[22] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *J. Med. Syst.*, vol. 42, no. 7, pp. 1–7, Jul. 2018.
[23] K. R. Ozyilmaz and A. Yurdakul, "Designing a blockchain-based IoT with Ethereum, Swarm, and LoRa: The software solution to create high availability with minimal security risks," *IEEE Consum. Electron. Mag.*, vol. 8, no. 2, pp. 28–34, Mar. 2019.
[24] P. Gangwani, A. Perez-Pons, T. Bhardwaj, H. Upadhyay, S. Joshi, and L. Lagos, "Securing environmental IoT data using masked authentication messaging protocol in a DAG-based blockchain: IOTA tangle," *Future Internet*, vol. 13, no. 12, p. 312, Dec. 2021.
[25] U. Khalid, M. Asim, T. Baker, P. C. K. Hung, M. A. Tariq, and L. Rafferty, "A decentralized lightweight blockchain-based authentication mechanism for IoT systems," *Cluster Comput.*, vol. 23, no. 3, pp. 2067–2087, Sep. 2020.
[26] B. K. Mohanta, U. Satapathy, S. S. Panda, and D. Jena, "A novel approach to solve security and privacy issues for IoT applications using blockchain," in *Proc. Int. Conf. Inf. Technol. (ICIT)*, Dec. 2019, pp. 394–399.
[27] S. S. Panda, U. Satapathy, B. K. Mohanta, D. Jena, and D. Gountia, "A blockchain based decentralized authentication framework for resource constrained IoT devices," in *Proc. 10th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2019, pp. 1–6.
[28] J. Wan, J. Li, M. Imran, D. Li, and Fazal-e-Amin, "A blockchain-based solution for enhancing security and privacy in smart factory," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3652–3660, Jun. 2019.
[29] B. K. Mohanta, D. Jena, S. Ramasubbareddy, M. Daneshmand, and A. H. Gandomi, "Addressing security and privacy issues of IoT using blockchain technology," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 881–888, Jan. 2021.
[30] M. T. Hammi, B. Hammi, P. Bellot, and A. Serhrouchni, "Bubbles of trust: A decentralized blockchain-based authentication system for IoT," *Comput. Secur.*, vol. 78, pp. 126–142, Sep. 2018.
[31] A. Z. Ourad, B. Belgacem, and K. Salah, "Using blockchain for IoT access control and authentication management," in *Proc. 3rd Int. Conf., Held Part Services Conf. Fed. Internet Things (ICIOT)*, Seattle, WA, USA. Cham, Switzerland: Springer, Jun. 2018, pp. 150–164.
[32] S.-C. Cha, J.-F. Chen, C. Su, and K.-H. Yeh, "A blockchain connected gateway for BLE-based devices in the Internet of Things," *IEEE Access*, vol. 6, pp. 24639–24649, 2018.
[33] W. Zhao, I. M. Aldyaflah, P. Gangwani, S. Joshi, H. Upadhyay, and L. Lagos, "A blockchain-facilitated secure sensing data processing and logging system," *IEEE Access*, vol. 11, pp. 21712–21728, 2023.
[34] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 33–39, Aug. 2018.
[35] R. Li, T. Song, B. Mei, H. Li, X. Cheng, and L. Sun, "Blockchain for large-scale Internet of Things data storage and protection," *IEEE Trans. Services Comput.*, vol. 12, no. 5, pp. 762–771, Sep. 2019.
[36] P. Gangwani, A. Perez-Pons, S. Joshi, H. Upadhyay, and L. Lagos, "Integration of data science and IoT with blockchain for industry 4.0," in *Blockchain and Its Applications in Industry 4.0*. Singapore: Springer, 2023, pp. 139–177.
[37] G. Rathee, M. Balasaraswathi, K. P. Chandran, S. D. Gupta, and C. S. Boopathi, "A secure IoT sensors communication in industry 4.0 using blockchain technology," *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 1, pp. 533–545, Jan. 2021.
[38] S.-K. Kim, U.-M. Kim, and J.-H. Huh, "A study on improvement of blockchain application to overcome vulnerability of IoT multiplatform security," *Energies*, vol. 12, no. 3, p. 402, Jan. 2019.
[39] A. Al-Dulaimi, S. Al-Rubaye, and Q. Ni, "Energy efficiency using cloud management of LTE networks employing fronthaul and virtualized baseband processing pool," *IEEE Trans. Cloud Comput.*, vol. 7, no. 2, pp. 403–414, Apr. 2019.
[40] H. Vranken, "Sustainability of Bitcoin and blockchains," *Current Opinion Environ. Sustainability*, vol. 28, pp. 1–9, Oct. 2017.
[41] C. Zhang, C. Wu, and X. Wang, "Overview of blockchain consensus mechanism," in *Proc. 2nd Int. Conf. Big Data Eng.*, May 2020, pp. 7–12.
[42] Z. Xiang, D. Malkhi, K. Nayak, and L. Ren, "Strengthened fault tolerance in Byzantine fault tolerant replication," in *Proc. IEEE 41st Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2021, pp. 205–215.
[43] B. Podgorelec, M. Turkanović, and S. Karakatič, "A machine learning-based method for automated blockchain transaction signing including personalized anomaly detection," *Sensors*, vol. 20, no. 1, p. 147, Dec. 2019.

AVISHAEK DEEP received the bachelor's degree in electronics and communication engineering and the master's degree in applied artificial intelligence from the School of Aerospace, Transport, and Manufacturing, Cranfield University, U.K. He joined the Indian Naval Academy. His research interests include the IoT, AI, blockchain, and communications and networking.

ADOLFO PERRUSQUÍA (Member, IEEE) received the B.Eng. degree in mechatronic engineering from the National Polytechnic Institute (UPIITA-IPN), in 2014, and the M.S. and Ph.D. degrees in automatic control from the Automatic Control Department, CINVESTAV-IPN, in 2016 and 2020, respectively. He is currently a Lecturer with the School of Aerospace, Transport and Manufacturing, Cranfield University, and a former UK-IC Postdoctoral Research Fellow. His main research interests include robotics, mechanisms, machine learning, reinforcement learning, nonlinear control, system modeling, and system identification. He is a member of the IEEE Computational Intelligence Society.

LAMEES ALJABURI is currently pursuing the master's degree in computer engineering with Near East University, Cyprus, which reflects her commitment to furthering her education and research to focus on AI, ML, security, blockchain, and communications and networking. She is a dedicated professional with a strong background in computer engineering and five years of valuable industrial experience.

SABA AL-RUBAYE (Life Senior Member, IEEE) is currently the Chair Professor of telecommunications and autonomous systems with the School of Aerospace, Transport, and Manufacturing, Cranfield University. She is an honored member of the U.K. Government Telecoms Innovation Network (UKTIN), actively contributing to the advancement of the 6G ecosystem and cutting-edge technologies to drive innovation within the telecommunications industry and exploring its potential applications in future communication networks. She actively contributes to the development of industry standards, serving as a Voting Member for the IEEE P1920.2 Standard for Vehicle-to-Vehicle Communications for Unmanned Aircraft Systems. She is a member of IET. She is registered as a Chartered Engineer (C.Eng.).

WEISI GUO (Senior Member, IEEE) received the M.Eng. degree in engineering and the M.A. and Ph.D. degrees in computer science from the University of Cambridge, Cambridge, U.K., in 2005, 2011, and 2011, respectively.

He was a Turing Fellow with the Alan Turing Institute. He is currently a Chair Professor of human-machine intelligence with Cranfield University, Cranfield, U.K. He has published more than 180 articles and is a PI on a number of molecular communication research grants. His research has won him several international awards.