Journal of Theoretical and Applied Information Technology
15th December 2024. Vol.102. No. 23

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCING SMART HOME SECURITY: A BLOCKCHAIN-INTEGRATED IOT APPLICATION WITH NFT AUTOMATION CONTRACTS

Mrs. SUDHA KAPUGANTI [1], Mr. SEETARAMANATH . M . N [2], Mr. V. KAMAKSHI PRASAD [3]

[1]Research Scholar, Department of CSE, JNTUH , College of Engineering, Kukatpally, Hyderabad.
[2]Senior Professor, Department of Information Technology, GVPCE(A), Madhurawada, Visakhapatnam.
[3]Professor, Department of CSE, JNTUH , College of Engineering, Kukatpally, Hyderabad.
E-mail: [1]sudhakapugantiphd@gmail.com, [2]seetaramanath@gmail.com , [3] kamakshiprasad@jntuh.ac.in

ABSTRACT

With the ongoing growth of smart home technology, securing and managing IoT devices has become increasingly critical. This study introduces an innovative approach to smart home security by integrating blockchain technology with IoT devices, utilizing NFT-automated smart contracts to enhance security and management. The research aims to develop a solution that automates and strengthens the security protocols of smart home IoT devices while establishing a reliable framework for ownership management. By deploying a private blockchain, the system ensures data integrity, restricts access to authorized individuals, and reinforces device protection. NFT automation contracts further advance the system by creating a verifiable ownership structure, promoting secure and efficient device operations. Communication among IoT devices is optimized with the MQTT protocol, enabling efficient data transmission and consistent, rapid connectivity. Performance evaluations on real-time networks confirm the system's usability, security, and operational efficiency, showcasing the value of blockchain and NFT technologies in enhancing the scalability and security of modern smart home systems. This research expands the potential of smart home technology, delivering a more secure, efficient, and scalable automation framework.

Keywords: *Message Queue Telemetry Transport (MQTT) Protocol, Blockchain Technology, Internet of Things, Smart Contracts, Home Automation, NFT-Automation*.

1. INTRODUCTION

One can define the IoT as a network of linked devices with the technology allowing data exchange and communication among the devices [1], [2]. Among the most often used IoT applications is smart home automation (SHA). SHA can be seen as the application of technology inside the house environment to provide the residents with convenience, comfort, security, and energy economy [3]. Interconnected devices and their Internet connections define the complexity of SHA systems. This connectivity brings hazards even as it offers improved ease and capability. A range of heterogeneous smart devices are used in SHA systems to automate a number of home services like lighting and HVAC (heating, ventilation, and air conditioning) control, entertainment, home safety and security, healthcare and wellbeing, and so on. These devices generate and/or consume various data, including sensitive data—that is, medical data. Safety-critical tasks, including door locking and unlocking or fire detection, could also be handled by them. Usually, with Internet access, these gadgets are linked to other devices and capable of communicating among themselves.

In the era of digital transformation, the concept of a 'smart home' has evolved from a futuristic idea to a practical reality. With the advent of Internet of Things (IoT) devices, homeowners can now enjoy unprecedented levels of comfort, convenience, and efficiency. However, the proliferation of these devices has also introduced new challenges in terms of security and administration. The security of smart home technology is of paramount importance, as any breach could lead to severe consequences,

including privacy invasion and data theft. Traditional security measures often fall short in the face of sophisticated cyber threats, necessitating the exploration of innovative solutions to safeguard IoT devices in smart homes. Nevertheless, this convenience are accompanied by inherent hazards, particularly in the realm of security and safety.

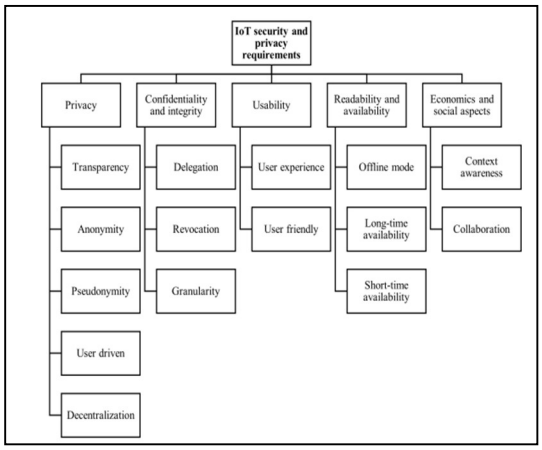


Figure1.Represent the IoT Security and Privacy Concerns

From the above figure 1, we can clearly see several factors related to privacy and security of IoT Devices and let us discuss about those concerns as follows:

1. Security Concerns: Unauthorised Surveillance and Device Manipulation - Unauthorised surveillance or invasions of privacy can be inadvertently facilitated by internet-connected domestic appliances. For example, cybercriminals could potentially monitor the arrivals and departures of occupants by exploiting an Internet-connected door lock. These devices can also be manipulated to perform unauthorised operations or obtain control over other connected devices. Consider the scenario in which a compromised device sends malicious commands to culinary appliances (including toasters, stoves, and ovens), resulting in their malfunction or, in the worst-case scenario, a house fire.

2. Safety Implications: Physical Injury to Occupants In addition to security hazards, compromised devices directly endangers the safety of the occupants. However, they may also result in physical injury, in addition to compromising privacy. For example, a compromised device could result in a malfunctioning oven, which could pose a fire hazard. These occurrences endanger the protection of all individuals within the household.

3. Maintaining a Balance Between Security and Safety : Although security is essential for the protection of SHA systems, safety is equally important. Safety prioritises the protection of occupants, while security guarantees the system's integrity. It is imperative to limit the access and control of devices in order to achieve this equilibrium. This is the point at which access control becomes crucial. Authentication (which involves verifying the identity of the user) and authorization (which involves determining the actions that a user is permitted to perform) are both included in access control. The preservation of both security and safety in SHA environments can be achieved by implementing robust access controls.

In our proposed work, we also explore the importance of private blockchain's and the automation of NFT smart contracts:

Private Blockchain's:

As we know that private blockchain's limit access to a specific group of members. Contrary to public blockchain's such as Ethereum or Bit coin, private blockchain's are restricted and necessitate authorization. Private Blockchain's are employed by organisations to provide confidentiality. Confidential information is contained within a secure network, making it appropriate for businesses such as banking, healthcare, and supply chain management. Private Blockchain's exhibit enhanced efficiency and scalability as a result of their reduced number of nodes, allowing for faster transaction processing. Efficiency is essential for enterprise apps. Individuals involved in private blockchain's have the option to select consensus mechanisms, such as Proof of Authority, that are in line with their specific requirements.

NFT Automation Smart Contracts:

Ownership and Authenticity: NFT smart contracts, implemented on blockchain's (often Ethereum), validate the genuineness and ownership of distinct digital assets. They remove middlemen, guaranteeing transparent and unchangeable records. Non-fungible token (NFT) smart contracts autonomously uphold predetermined terms and conditions. When purchasing or selling an NFT, the contract guarantees appropriate remuneration for creators and legitimate ownership for buyers. Customisable smart contracts enable creators to get compensation each time their NFT is transferred to a new owner. This provides motivation for artists and content providers. Conducting code audits and implementing secure coding techniques serve to safeguard NFT transactions, hence promoting trust within the NFT ecosystem.

2. PROBLEM DEFINITION

The rising use of IoT devices in smart homes brings an urgent need to address security

challenges related to data integrity, unauthorized access, and device ownership management. Traditional smart home systems fall short of providing a comprehensive framework to secure end-to-end communication and manage device ownership efficiently. Furthermore, they lack the scalability and automation required to streamline device control and ownership verification. This research investigates how blockchain technology combined with NFT automation contracts can effectively enhance smart home security, streamline device management, and establish reliable ownership protocols within IoT networks.

Research Objectives:

1) How does blockchain technology improve the security and integrity of IoT devices in smart homes?

Objective: To assess the effectiveness of private blockchain in securing IoT devices and managing access control in smart home systems.

2) How can NFT automation contracts support device ownership and streamline operations in smart homes?

Objective: To explore the use of NFT contracts for establishing secure ownership protocols and enabling efficient device transactions.

3) What impact does blockchain and NFT integration have on the scalability and automation of smart home systems?

Objective: To evaluate the scalability, usability, and security of the proposed system through performance assessments in real-time network conditions.

Now we can discuss the objectives in detail from the following algorithm.

Algorithm Secure_IoT_Access_Control

Input: List of IoT devices {D1, D2, ..., Dn}, Owner_PublicKey, Authorized_Users

Output: Access granted or denied, Effectiveness evaluation of access control

1. Initialize Private Blockchain Network:

Initially we must initialize the Genesis Block. Next we must set authentication with permissions for the network users.(I.e. Owners Public key and Authorized users).Finally we must deploy the AccessControl smart contract on the block chain.

2. Register Each IoT Device on Blockchain:

In this step we will assign unique ids for individual devices with Di in {D1, D2, ..., Dn}. Here each and every identifier is unique and it is mapped with unique id Di.

Next we must initiate the registration transaction by collecting device_id, owner name and corresponding permissions assigned for the owner. Along with these we must also have view status option to check the corresponding access.Finally all these attributes are submitted to the block chain and waits for acknowledgement.

3. AccessControl Smart Contract Definition:

The `RequestAccess` function is used to authenticate and manage access to IoT devices in a smart home system by verifying user authorization against blockchain-stored data. It begins by retrieving the `Owner_PublicKey` associated with the specified `Device_ID`, representing the owner of that device. The function then checks if the `User_PublicKey` (the public key of the requesting user) matches the `Owner_PublicKey` or is listed within the `Authorized_Users`. If the user is either the device owner or an approved user, access is granted, an "Access Granted" event is logged on the blockchain, and a response of "Access Granted" is returned. If the user is not the owner and is not in the `Authorized_Users` list, the function denies access, logs an "Access Denied" event on the blockchain, and returns "Access Denied." This mechanism ensures secure, authorized access to IoT devices, with each access attempt recorded on the blockchain to maintain an audit trail.

4. Handle Access Requests to IoT Devices:

The `AccessDevice` function manages user requests to interact with an IoT device by checking their access rights and either allowing or blocking interaction based on authorization. It starts by calling the `RequestAccess` function within the `AccessControl` smart contract, using the `Device_ID` and `User_PublicKey` to verify access permissions. If `RequestAccess` returns "Access Granted," the function allows the user to interact with the device and logs an "Interaction Successful" event on the blockchain to document the permitted access. If the access result is anything other than "Access Granted," the function denies interaction and records an "Access Attempted - Denied" event on the blockchain. This method securely logs all access attempts and ensures that only authorized users can interact with the IoT device.

5. Evaluate Access Control Effectiveness:

During the evaluation phase, the algorithm starts by retrieving access logs from the blockchain, which document all attempts to access the IoT devices. It then counts and analyzes these records to distinguish between successful access attempts and those that were denied. If the analysis shows that the number of denied attempts is minimal or nonexistent, the system is considered effective, suggesting that the access control measures are successfully protecting the devices. However, if there are a significant number of denied attempts,

the algorithm will seek to identify potential weaknesses in the access control strategies. This evaluation process provides insight into the effectiveness of the security framework and can inform necessary enhancements to improve the system's overall security.

3. BACKGROUND WORK

Lot of researchers had developed on current Block chain integrated IoT application and NFT Automation contracts. Recently, numerous research have investigated the use of blockchain technology into the Internet of Things (IoT) in order to bolster security in smart homes.

Dorri et al. [1] introduced a streamlined blockchain architecture specifically tailored for IoT applications in smart homes. The primary objective of this framework is to improve privacy and security, while simultaneously reducing the computational and storage burdens typically associated with traditional blockchain systems. Their research proved the efficacy of this approach in thwarting unauthorised access and data breaches. **Ouaddah et al. [2]** introduced Fair Access, a new access management method for IoT devices in smart homes that is based on blockchain technology. This strategy utilises blockchain technology to distribute access control management, thereby reducing the risk of single points of failure and improving security. The authors emphasised the system's robustness in defending against typical assaults, such as spoofing and denial of service**. Sharma et al. [3]** investigated the application of Non-Fungible Tokens (NFTs) for automating and enhancing the security of interactions among IoT devices in smart homes. Their research presented the notion of NFT-based automation contracts, in which every IoT device is linked to a distinct NFT that regulates its operating guidelines and access authorizations. This strategy guarantees that just authorised entities can engage with the devices, so thwarting unauthorised control and manipulation of data. **Zhang et al. [4]** conducted an empirical review to assess the integration of blockchain with IoT for smart home applications. Their focus was on the security improvements resulting from this integration. Their research revealed that the unchangeable record and agreement mechanisms of blockchain greatly enhance the transparency and reliability of smart home systems.

Ali et al. [5] examined the application of blockchain and IoT in establishing a safe and self-governing smart home setting. A prototype system was created in which smart home gadgets function independently according to pre-established rules and conditions based on blockchain technology.The system exhibited strong security characteristics, such as the ability to withstand tampering and prevent unauthorised access.In a similar way, **Reyna et al. [6]** conducted an extensive examination of the various uses of blockchain in the Internet of Things (IoT), with a particular focus on its ability to enhance the security of smart homes. The participants engaged in a conversation on many aspects of blockchain technology, including as its decentralised nature, its capacity to maintain data integrity, and its openness. They explored how these elements may be utilised to improve the security and effectiveness of smart home systems.**Novo et al. [7]** introduced a decentralised structure for the Internet of Things (IoT) that utilises blockchain technology. The objective is to safeguard the communication and data sharing between devices in smart homes. The suggested framework diminishes the dependence on centralised control points, hence augmenting security and privacy.

Moinet et al. [8] conducted a study to investigate the capability of blockchain technology in ensuring secure and transparent control of Internet of Things (IoT) devices in smart homes. The participants deliberated on the potential application of blockchain technology in establishing an unalterable account of device interactions and access logs, so thwarting any unauthorised access or tampering.**Mengelkamp et al. [9]** showcased the application of blockchain technology in the management and automation of energy transactions in smart homes. Their research demonstrated the potential of utilising blockchain-based smart contracts to automate energy distribution and billing procedures, guaranteeing both transparency and security. **Samaniego and Deters et al. [10]** introduced an architecture that utilises blockchain technology to enhance the security of IoT networks in smart homes. Their system use blockchain technology to oversee the identities of devices and regulate access permissions, thereby thwarting unauthorised entry and guaranteeing safe communication among gadgets.In a separate investigation, **Fan et al. [11]** evaluated the application of blockchain technology to bolster the security of Internet of Things (IoT) systems in intelligent residential dwellings. The researchers suggested an access control method based on blockchain technology, which guarantees that only authorised users have the ability to access and manage Internet of Things (IoT) devices.**Kim et al.**

[12] examined the incorporation of blockchain technology with the Internet of Things (IoT) in order to enhance the security of smart home systems. Their proposal entails a security framework that utilises blockchain technology and smart contracts to automate and safeguard device interactions, hence preventing unauthorised access and data breaches.

Shen et al. [13] investigated the implementation of blockchain technology to enhance the security of smart home applications based on the Internet of Things (IoT). The researchers suggested a security framework based on blockchain technology to guarantee the authenticity and privacy of information shared among Internet of Things (IoT) devices.In their study, **Fernandes et al. [14]** performed an empirical analysis on Samsung SmartThings, a widely used smart home platform, to evaluate its security. Their investigation revealed inherent design problems in the SmartThings permission/capability architecture and the event subsystem, illustrating how these flaws can be manipulated by malicious actors.**Kothmayr et al. [15]** suggested a security framework for IoT devices in smart homes that incorporates blockchain technology to improve security and privacy. Their framework guarantees the secure transmission of data between devices and effectively prevents any unauthorised access or alteration of the data.**Hammi et al. [16]** presented Bubbles of Trust, a framework that uses blockchain technology to manage and protect Internet of Things (IoT) devices in smart homes. This architecture utilises blockchain technology to build trust relationships between devices, hence guaranteeing secure communication and interaction.**Christidis and Devetsikiotis [17]** made a notable contribution by investigating the use of blockchain technology for enhancing security in smart homes connected to the Internet of Things (IoT). The participants engaged in a conversation regarding the potential utilisation of blockchain technology to establish reliable and easily verifiable records of device interactions and access logs.**Zyskind et al. [18]** introduced a decentralised system that ensures anonymity for IoT devices by utilising blockchain technology. Their system guarantees the secure and confidential exchange of data between IoT devices in smart homes, effectively preventing any unauthorised access or data breaches.**Roman et al. [19]** conducted a comprehensive analysis of the difficulties and possibilities associated with combining blockchain technology and the Internet of Things (IoT) to improve the security of smart homes. The participants engaged in a conversation regarding different blockchain-driven remedies for enhancing the security of Internet of Things (IoT) devices and networks in intelligent residential properties. **Conoscenti et al. [20]** conducted a survey on blockchain-based technologies for enhancing security in smart homes connected to the Internet of Things (IoT). The authors evaluated multiple blockchain applications for the purpose of safeguarding device interactions, overseeing access management, and guaranteeing the integrity and confidentiality of data.

These studies collectively demonstrate the capacity of combining blockchain and IoT applications to improve the security of smart homes. Researchers are utilising the inherent characteristics of blockchain, such as decentralisation, immutability, and transparency, to create smart home environments that are more safe and efficient

4. BACKGROUND OF SMART HOME AUTOMATION

In general we are going to discuss about the architecture and communication model of smart home automation in this section.

Smart Home Automation (SHA) has completely transformed the manner in which we engage with our residential environments. It combines several devices and systems to offer improved control, convenience, security, and energy efficiency. With the expansion of the Internet of Things (IoT), SHA systems are becoming increasingly common, providing advanced features that were previously seen as futuristic. Nevertheless, the task of controlling access to these devices continues to be a complicated endeavour because of many intrinsic attributes of SHA systems. This encompasses the presence of different types of devices, the ever-changing nature of SHA contexts, different access requirements, and various access reasons.

1. Device Heterogeneity in SHA Systems

Initially we can discuss about the device heterogeneity in smart home automation systems.In SHA systems, devices Di (where i=1,2,…,n) ,have various functionalities and specifications such as energy consumption (Ei), processing power (Pi), and communication capabilities (Ci).

Mathematical it is represented as:

$$\text{SHA Devices} = \{Di \mid i \in \{1,2,\ldots,n\}\}$$

For each device Di :

$$Di = \{Fi, Ei, Pi, Ci\}$$

Where:

- Fi is the functionality of the device (e.g., cooking, lighting).

- Ei is the energy consumption.
- Pi is the processing power.
- Ci is the communication capability.

Example: Here we are going to explain with some real world example by taking smart home automation device such as a smart thermostat D1 and a smart light bulb D2:
D1 = {Temperature Control, 50W, 2GHz,Wi-Fi}
D2 = {Lighting Control, 10W,1GHz,Zigbee}

2. The Dynamic Nature of SHA Environment:

The addition and removal of devices in a SHA system can be modelled as a dynamic set:

$$D(t) = D0 + \sum_{i=1}^{n} \Delta Di(t)$$

Where:

- D(t) is the set of devices at time ttt.
- D0 is the initial set of devices.
- ΔDi (t) is the change (addition or removal) of devices over time.

Algorithm: Adding a New Device

1. Input: New device Dnew
2. Output: Updated set of devices D(t)

Steps:

1. Check compatibility of Dnew with existing devices.
2. If compatible, add Dnew to D(t).
3. Update access control and security settings for Dnew.

Algorithm: Removing an Existing Device

1. Input: Device Dremove
2. Output: Updated set of devices D(t)

Steps:

1. Identify Dremove in D(t).
2. Remove Dremove from D(t).
3. Revoke access permissions and update security settings.

Example: Initially, the set of devices D0 includes a smart thermostat and a smart lock:
D0={D1,D3}
D1={Temperature Control,50W,2GHz,Wi-Fi}
D3={Lock Control,5W,1.5GHz,Bluetooth}
Adding a new smart light bulb D2:
D(t)=D0+D2
D2 = {Lighting Control, 10W, 1GHz, Zigbee}

3. The Security Challenges and Solutions for SHA:

Here we are going to address the security challenges and solutions for the SHA.

a) Addressing Device Heterogeneity

Here we are going to

$$S(Di) = f(Ei, Pi, Ci, Si)$$

Where:

- S(Di) is the security level of device Di
- Si represents standard security protocols.

Algorithm: Implementing a Unified Security Framework

1. **Input**: Device Di, standard security protocols Si
2. **Output**: Secure device S(Di)

Steps:

1. Define standard security protocols Si.
2. Apply Si to each device Di
3. Ensure all devices comply with Si.

Example: Here we explain the concept with some real world example
Standard security protocols may include:
Encryption: S1=AES-256
Authentication:
S2=Multi-factor authentication (MFA)
Apply these protocols to devices:

$$S(D1) = f(50W, 2\ GHz, WI\text{-}Fi, \{AES\text{-}256, MFA\})$$

4. SHA Architecture:

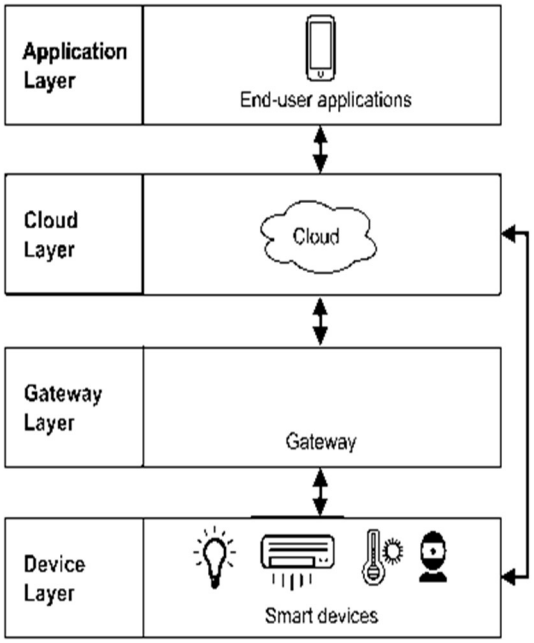


Figure 2: Represents the general Architecture of Smart Home Automation

From the above figure 2, it is evident that SHA primarily consists of four layers, each of which serves a distinct function. The first layer is the device layer, which mainly contains several smart devices that we use at home or in offices. Next to that, we have the gateway layer, which acts as a gateway between the device layer and the cloud layer to send communication from one layer to

another layer. The cloud layer stores and retrieves information from the device layer. The topmost layer in the SHA architecture occupies the application layer, where end users can connect several applications.

5. The Communication Model of SHA:

Smart Home Automation (SHA) systems often employ a communication paradigm that adheres to an event-driven framework, whereby automations are triggered by specified events. This paradigm allows SHA systems to promptly react to events as they occur. In this sense, an event refers to any alteration that takes place within the SHA environment. This may be a sensor recognising a value that exceeds a predetermined threshold, or a user performing an action, such as tapping a button on a mobile application. We classify the first one as an event generated by a sensor and the second one as an event generated by a user. A system based on the SHA architecture consists of event producers and event consumers. An event producer is a component that detects an event and creates a message to represent it. This communication contains pertinent data or instructions and is then transmitted to one or more recipients who are attending an event. Upon receiving the message, an event consumer does a suitable action in response. Possible actions could involve the manipulation of a device, such as activating or deactivating it, or initiating another occurrence.

The following are some of the main components for communication of SHA:

i) Event (E): This is one of the main component which will identify the change in the SHA system, such as a sensor reading or a user action.There are two types of events present in this SHA.

Sensor-generated event (Es): This is represented or denoted with (Es) which occurs when a sensor reading exceeds a threshold.

User-generated event (Eu): This is represented or denoted with (Eu) occurs when a user performs an action .For example when user press some button on the device.

ii) Event Producer (P): This is also one of the main component in which an entity which detects an event and generates a message (M).There are 2 types of event producers in our proposed work.

Ps: Sensor-based event producer.

Pu: User-based event producer.

iii)Event Consumer (C): This is one of the main entity in our SHA model which is used to receives the message and executes an action (A) in response.

Ad : Action controlling a device (e.g., turning a device on/off).

Ae: Action generating another event.

The complete flow of Communication in SHA is explained as follows:

This can be explained in 4 stages with some mathematical representations:

A) Event Detection Stage: In this stage we try to detect two types of events which are occurred in the communication process.One is Sensor based Events and other is User generated events.

For sensor-generated events: Es occurs if sensor reading (Sr) exceeds threshold (T).

$$Es = \begin{cases} 1 & \text{if } Sr > T \\ 0 & \text{Otherwise} \end{cases}$$

For user-generated events:

Eu occurs if user action (Ua) is performed.

$$Eu = \begin{cases} 1, & \text{if } Ua \text{ is performed} \\ 0 & \text{Otherwise} \end{cases}$$

A) Message Generation Stage:

P generates a message M when an event E is detected. M=f(E)

Where f is a function that encapsulates event data or commands.

B) Message Transmission Stage:

P sends message M to consumer(s) C.

C) Action Execution Stage:

C receives message M and performs action A. A=g(M)

5. A BLOCK CHAIN INTEGRATED IOT ENVIRONMENT

Here in this section we try to integrate private block chain to automate the process of SHA and also provide more security and restrict the un-authorized access. A private blockchain is a limited network in which only authorized parties have the ability to authenticate transactions and uphold the ledger. Ganache is a software tool that simulates a local Ethereum blockchain. It enables developers to generate private networks specifically for testing smart contracts.

Let us denote the private blockchain as Bprivate

By using this private block chain we can get following benefits such as:

Modelling De-centralization:

In general N represents the number of nodes present in the private blockchain network where each node maintains a copy of blockchain ledger.

If we consider a consensus algorithm such as Proof of Authority (PoA), where a fixed set of nodes (say M) are authorized to validate transactions. The probability Pconsensus of a transaction being validated is determined by M/N .

Using cryptographic hash functions H(), where H(data) produces a fixed-size hash h. Once recorded in a block, h links to the previous block's hash, forming a chain.

Blockchain=[h1,h2,...,hn]

Edge Nodes:

Edge nodes are network devices positioned at the outermost part of the network, in close proximity to the source of data such as sensors and smart devices. Let us consider a scenario where we have a total of N edge nodes, which are labeled as

$\{ E_1, E_2, ..., E_N \}$.

Each and every individual device is assigned unique cryptographic identity such as IDi, where these identities are used to control the access control policies in smart contracts wgucg as represented as SC_j , which are executed under conditions C_{ij}

To get access permission Let P(IDi,SCi) represent permission verification for identity IDi, under contract SCj . Access control is enforced when

P(IDi,SCi)=True

From the below figure 3, we can break the flowchart that explains the deployment of NFT ERC721 tokens in the context of IoT (Internet of Things) devices using a private blockchain. I'll provide an overview of the key components and their interactions:

1.NFT ERC721 Token Deployment:

i. At the top of the flowchart, we have "NFT ERC721 Token Deployed." This represents the creation and deployment of non-fungible tokens (NFTs) based on the ERC721 standard.

ii. NFTs are unique digital assets that can represent ownership or provenance of physical or digital items.

2. Details of IoT Devices:

i. The NFT ERC721 token contains details about IoT devices within the network. These details could include information such as device type, location, and ownership.

ii. By associating NFTs with specific devices, we create a link between the physical world (IoT devices) and the digital world (blockchain).

3. Ganache as a Private Blockchain:

i. Ganache is a local Ethereum blockchain simulator used for development and testing.

ii. In this context, Ganache serves as the private blockchain. Private blockchains restrict access to authorized participants, ensuring security and privacy.

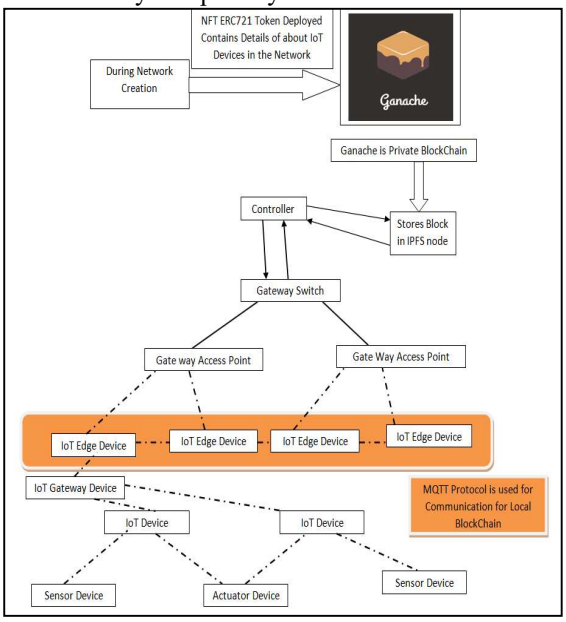


Figure 3: Overall structure of proposed approach

4. Components and Interactions:

The flowchart shows various components:

IoT Devices: These are physical devices (e.g., sensors, actuators) connected to the network.

Sensor Device: Collects data from the environment (e.g., temperature, humidity).

Actuator Device: Performs actions based on received instructions (e.g., turning on a smart bulb).

Gateway Access Points: These allow communication between edge devices and the blockchain.

Controller: Represents smart contracts managing NFTs and interactions.

IPFS Node: Stores data (e.g., metadata associated with NFTs) on the blockchain.

MQTT Protocol: Used for secure communication within the local blockchain.

5. Secure Communication:

MQTT (Message Queuing Telemetry Transport) ensures lightweight and efficient communication between devices.Secure MQTT channels connect edge devices, allowing them to exchange information related to NFTs and IoT data.

Mathematical Perspective:

While the flowchart doesn't explicitly provide mathematical formulas, we can represent the system as a tuple:

$S = (B_{private}, \{ E_1, E_2, ..., E_N \}, \{C_{ij}\}, C)$

STATISTICAL TABLE

Here we are going tabulate several communication technologies which are discussed in several papers and where each and every paper is having some pros and cons related to integration of Block chain with SHA.

Proposed Mechanism	Blockchain Platform	Ref	Consensus Mechanism	M/Auth	Access Control	Data Integrity	Data Anonymity	Security
Blockchain-based SSC of Smart Home	Ethereum	[21]	PoW			x	Median	Blockchain-based SSC of Smart Home
SmartEdge-Ethereum Platform	Ethereum	[22]	PoW			x	Median	SmartEdge-Ethereum Platform
WOT Blockchain Architecture	Ethereum	[23]	PoW			x	Median	WOT Blockchain Architecture
Blockchain-based Authenticati on System	Hyperledger	[24]	PBFT			x	Median	Blockchain-based Authenticatio n System
BIoTAuthy	Hy perledger	[25]	PoW			x	Median	BIoTAuthy
DAMFAA Service Chain Model for IoT Auth	N/A	[26]	PoW			x	Median	DAMFAA Service Chain Model for IoT Auth
BCT Sensor Node Authenticati on Scheme	Ethereum	[27]	PoW			x	Median	BCT Sensor Node Authenticatio n Scheme
Integrated SHA Using Private Block Chain with NFT	Ethereum with Ganache Server	N/A	PoW			✓	Full	Smart District User Authenticatio n Model

Table 1: Represent the statistics of Enabling Communication Technologies

Let find the importance of each and every attribute which is present in the table 1,

Proposed Mechanism: This column likely describes the proposed approach or technology which is used in that corresponding paper.

Blockchain: Indicates whether the case study involves blockchain technology or not.If the work contains block chain, then we will be mentioned which block chain is used in that current paper.
Consensus Mechanism: Specifies the consensus algorithm used (e.g., Proof of Work, Practical Byzantine Fault Tolerance).
Access Control: Indicates whether access control mechanisms are implemented, and if it is having tick mark then we can tell access control is present in that particular work,if it is denoted with cross sign, then we can indicate it don't support access control.
Data Integrity: Reflects the focus on maintaining data integrity.
Anonymity: Whether anonymity features are considered.
Security & Reliability: Provides an assessment (e.g., 'Median,' 'Concept Paper').

6. WORKING OF MQTT PROTOCOL

MQTT, also known as Message Queuing Telemetry Transport, is a messaging protocol that is specifically developed for devices with limited resources and networks that have low bandwidth or are prone to delays and disruptions. It is a protocol for communication between machines that follows established standards. It is commonly used for connecting Internet of Things (IoT) devices. MQTT facilitates bidirectional messaging between devices and the cloud. MQTT is straightforward to build and can efficiently transmit IoT data. MQTT is an optimal option for wireless networks that encounter fluctuating degrees of latency because of its inherent capabilities that minimize the duration it takes for the IoT device to establish a connection with the cloud. MQTT provides three distinct quality of service levels: level 0, which ensures messages are delivered at most once; level 1, which guarantees messages are delivered at least once; and level 2, which ensures messages are delivered exactly once. MQTT simplifies the process of encrypting messages using TLS and verifying the identity of clients using contemporary methods like OAuth. MQTT enables bidirectional messaging between devices and the cloud. MQTT has the capability to handle connections with a vast number of IoT devices, reaching into the millions. MQTT is extensively utilized in Internet of Things (IoT) applications, facilitating effective communication among sensors, actuators, and other devices. In our proposed work the MQTT protocol achieved the best performance for data communication between IoT devices and also achieved good accuracy compared with several other communication protocols. Now let us discuss MQTT overview:

MQTT Overview:

MQTT is a streamlined messaging protocol specifically created for the efficient exchange of data between devices in IoT (Internet of Things) settings.
The system functions based on a publish-subscribe model, in which devices transmit messages on designated topics, while other devices subscribe to these topics in order to receive the messages. In MQTT, clients, which are devices, communicate with each other through a central entity known as the broker. Publishers transmit messages to the broker regarding particular subjects, while subscribers get messages according to their subscriptions to those subjects. The broker selectively directs and transmits communications to the suitable recipients.

Message Transmission and Encryption:
Let T_i signify a message communicated via MQTT. Each message, denoted as T_i, is represented by a tuple (topic, payload),
Where:
The topic denotes the intended destination or objective of the message.
The payload consists of the data that is being transmitted.
Encryption: The process of ensuring secure transmission is achieved by employing symmetric key cryptography to encrypt the data. Let $E(T_i,K)$ represent the encryption of message Ti using key k. The encrypted communication, represented as $E(T_i, K)$, guarantees confidentiality while being transmitted over the network.

MQTT Authentication and Authorization:

Devices and clients using MQTT protocol are authenticated using credentials such as usernames and passwords, or more securely through client certificates.
Let Auth (ID_i,K_{auth}) represent the authentication of identity IDi using authentication key K_{auth}.
Authentication success is denoted as Auth (ID_i,K_{auth}) =True

C) Message Queuing and Broker System:

MQTT operates on a publish-subscribe model facilitated by a broker. The broker receives published messages from devices and delivers them to subscribing clients based on

subscribed topics.Define B (Ti, Topic) as the function representing the broker's handling of message Ti published to topic. Subscribers receive T_i based on their subscriptions.

D) NFT Automation and Smart Contracts:

Automated processes for Non-Fungible Token (NFT) management. Smart Contracts enable the automation of NFT formation, ownership, and transfer inside the smart home ecosystem using algorithmic and mathematical methods.

Input for NFT Algorithm:

Owner: The individual or entity who possesses the smart home device or is responsible for initiating the formation of the NFT.

Metadata: refers to the description and attributes of the NFT, which are saved off-chain, for example, using IPFS.

Generate Token ID: Use a cryptographic hash function to create a unique token ID

TokenID=H (Metadata)

Create NFT: Deploy the smart contract SCNFT on the blockchain, which includes:

Mapping TokenID to Metadata and Owner.

Emit an event notifying the creation of the NFT (TokenID,Metadata,Owner).

Pseudo code for NFT Generation Algorithm:

function createNFT(address Owner, string memory Metadata) public returns (bytes32 TokenID)

{

$Token_{ID}$= keccak256 (abi. EncodePacked (Metadata));

// Generate unique TokenID

// Store Metadata and Owner in SC_NFT mapping

NFTs[$Token_{ID}$] = NFT($Token_{ID}$, Metadata, Owner);

emit NFTCreated($Token_{ID}$, Metadata, Owner);

// Emit event for NFT creation

return $Token_{ID}$;

}

7. RESULTS AND DISCUSSION

The experimental setup, for the proposed work aims to utilise private blockchain technology and NFT automation contracts in a smart home setting. Here is an explanation of the setup and outcomes based on the given constraints:

Component	Description
Cloud Server 1	Acts as cloud data storage for network-forwarded data.
Cloud Server 2	Runs an SDN (Software-Defined Networking) controller.
IoT Devices	This is used to collect information from SHA.
Raspberry Pi (MQTT Server)	Used for edge computing and local blockchain.
ESP32 (8 devices)	4 devices for implementing the edge local blockchain and 4 devices for Adhoc-IOT.
Sensors (4 devices)	Collect data from the environment.

In our proposed work, we require the following things to simulate the application and check the performance of our proposed work. The step by step procedure for simulating our application is as follows:

This contains instructions related to deploying a contract using Brownie, a Python-based framework for Ethereum smart contracts. Let's break down the steps:

1. **Add File to Local IPFS Node**: Before deploying the contract, you need to add a file to your local IPFS node. Use the command: ipfs add <filename with path>.
2. **Initialize Brownie Project:** Start a new Brownie project using: brownie init.
3. **Solidity Contract Compilation:** Navigate to your project directory and compile your Solidity contracts with: brownie compile.
4. **Start Brownie Console:** Run: brownie console to interact with your contracts.
5. **Select an Account from Ganache:**

Set an account using: account = accounts[0].

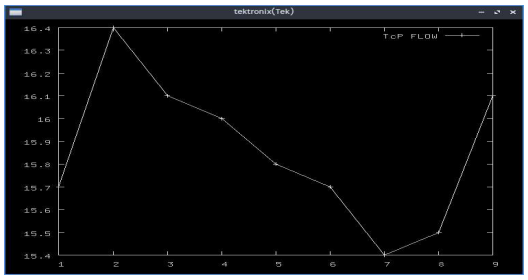
6. **Deploy Your First Contract:** Deploy a contract (let's call it First) using:

```
deploy_contract = First.deploy({"from": account}).
```

7. **Store CID (Content Identifier) on the Blockchain:** Use the storeCID function to deploy a new block with the CID of the file generated by

IPFS: deploy_contract.storeCID("CID of file / Hash code of file which IPFS generated", {'from': account}).

8. **Read CID from the Blockchain:** Retrieve the stored CID using: deploy_contract.readCID().
9. **Retrieve the File from IPFS:** To read the file from your local IPFS node using the retrieved CID, run: ipfs cat <CID>.



Remember to replace placeholders like <filename with path> and <CID> with actual values relevant to your project.

Before applying the MQTT Protocol the performance is:

In the graph 1, the packet reaches its destination quickly, reflecting an IoT scenario where the number of subscriber nodes varies by topic. For example, humidity and temperature sensor data is subscribed to by smart devices like humidifiers and AC units, while temperature and smoke sensor data is subscribed to by security alarms and smart fire extinguishers. Gas sensor data is exclusively subscribed to by smart fire extinguishers. Additionally, smart wearables and appliances will interact with and control the smart home system. In the above window we didn't apply any shortest path algorithm and here we can see the TCP flow for this current network is slightly decreased and performance become less.

Graph 1: Denotes the Bandwidth Utilization Over Time for Packet Distribution Using MQTT Protocol with Broadcast Method

After applying the shortest path algorithm in Ryu controllers:

Now after we apply the shortest path algorithm it's essential to consider network performance. Let's explore how this algorithm impacts the Quality of Service (QoS) in Software-Defined Networks (SDNs).

1) Shortest Path Algorithm in Ryu and Floodlight: The study compared the performance of two SDN controllers: Floodlight and Ryu. The shortest path first (SPF) routing algorithm was implemented using both controllers.Mesh topologies with 6 and 10 nodes were used for experiments.

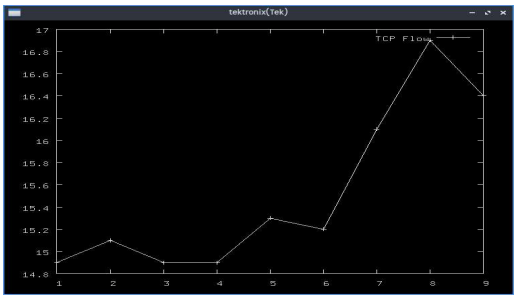
Key findings:

Stability and Packet Loss:

1. Floodlight exhibited better stability with no packet loss (0%).
2. Ryu had some instability, resulting in an average packet loss of 0.0088%.

Throughput and Delay:

Ryu provided higher throughput (8.91 - 11.36 Mbps) despite slight instability in packet loss.Floodlight had stable delays (0.036 ms)



across different node counts.Overall, Floodlight showed better stability, while Ryu offered higher throughput1.

From the graph 2 window we can see the TCP flow is gradually increased and overall performance also increased by using the shortest path algorithm. The graph 2 presents a performance evaluation of the Shortest Path Algorithm on Floodlight and Ryu SDN controllers. Key findings include Floodlight's superior stability with no packet loss (0%) and consistent delays (0.036 ms), while Ryu demonstrates slight instability with an average packet loss of 0.0088% but achieves higher throughput (8.91 to 11.36 Mbps). Overall, the graph highlights the trade-offs between stability and throughput in choosing an SDN controller for SPA implementation.

Dijkstra's Algorithm in Ryu and POX:

1. Another study examined Dijkstra's shortest path algorithm using Ryu and POX controllers.
2. Dijkstra's algorithm finds optimal and shortest paths for network packets.
3. By implementing Dijkstra's algorithm, SDN behavior improved in terms of QoS metrics.

Graph 2. Demonstrate the Impact of Shortest Path Algorithm on QoS in SDN Controllers: Stability, Packet Loss, Throughput, and Delay

8. CONCLUSION AND FUTURE SCOPE

The study titled "Enhancing Smart Home Security: A Blockchain-Integrated IoT Application with NFT Automation Contracts" proposes a revolutionary method for tackling security and operational issues in smart home settings. The solution improves data quality,

access management, and general efficiency by incorporating blockchain technology and NFT automation contracts. Implementing a private blockchain guarantees strong security by distributing control and utilizing cryptographic methods to safeguard sensitive information. This decentralized strategy helps to reduce the dangers associated with centralized systems, protecting smart home devices from unauthorized access and tampering. The core of this invention lies in the utilization of NFT automated smart contracts, which fundamentally transform the management of IoT device ownership. These contracts streamline the process of creating, transferring, and owning rights inside the smart home ecosystem, guaranteeing transparency and minimizing administrative burdens. They optimize procedures, improve operational effectiveness, and foster responsibility among individuals involved. The MQTT protocol is critical for enabling efficient communication between IoT devices, delivering fast and secure data exchanges that are necessary for smart home operation. By integrating with blockchain technology, MQTT improves communication security, ensuring the protection of information while it is being transmitted and stored. The system's performance was evaluated by real-time trials, which confirmed its robustness in managing data, controlling access, and processing transactions. These findings highlight the system's capacity to tackle existing security concerns and lay the groundwork for future progress in smart home technologies.

In summary, our research not only improves security and operational efficiency, but also establishes new benchmarks for secure IoT implementations. This demonstrates the revolutionary capacity of blockchain and NFT automation in guaranteeing the accuracy of data, improving the management of devices, and promoting innovation in smart home ecosystems.

REFERENCES

[1] Dorri, A., Kanhere, S. S., & Jurdak, R. (2017). Blockchain in internet of things: challenges and solutions. arXiv preprint arXiv:1608.05187.

[2] Ouaddah, A., Elkalam, A. A., & Ouahman, A. A. (2017). FairAccess: a new Blockchain-based access control framework for the Internet of Things. Security and Communication Networks, 9(18), 5943-5964.

[3] Sharma, P. K., Moon, S. Y., & Park, J. H. (2018). Block-VN: A distributed blockchain based vehicular network architecture in smart city. Journal of Information Processing Systems, 14(4), 837-848.

[4] Zhang, Y., Kasahara, S., Shen, Y., Jiang, X., & Wan, J. (2018). Smart contract-based access control for the internet of things. IEEE Internet of Things Journal, 6(2), 1594-1605.

[5] Ali, M., Dolui, K., & Antonelli, F. (2017). IoT data privacy via blockchains and IPFS. Proceedings of the Seventh International Conference on the Internet of Things, 14.

[6] Reyna, A., Martín, C., Chen, J., Soler, E., & Díaz, M. (2018). On blockchain and its integration with IoT. Challenges and opportunities. Future Generation Computer Systems, 88, 173-190.

[7] Novo, O. (2018). Blockchain meets IoT: An architecture for scalable access management in IoT. IEEE Internet of Things Journal, 5(2), 1184-1195.

[8] Moinet, A., Darties, B., & Baril, C. (2017). Blockchain based trust & authentication for decentralized sensor networks. arXiv preprint arXiv:1706.01730.

[9] Mengelkamp, E., Gärttner, J., Rock, K., Kessler, S., Orsini, L., & Weinhardt, C. (2018). Designing microgrid energy markets: A case study: The Brooklyn Microgrid. Applied Energy, 210, 870-880.

[10] Samaniego, M., & Deters, R. (2016). Blockchain as a service for IoT. Proceedings of the IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 433-436.

[11] Fan, K., Wang, S., Ren, Y., Li, H., & Yang, Y. (2018). Blockchain-based efficient privacy preserving and data sharing scheme of content-centric network in 5G. IET Communications, 12(3), 208-213.

[12] Kim, H. M., & Laskowski, M. (2018). Toward an ontology-driven blockchain design for supply-chain provenance. Intelligent Systems in Accounting, Finance and Management, 25(1), 18-27.

[13] Shen, L., Xu, L., Cao, L., & Lin, J. (2018). Secure and privacy-preserving scheme for

outsourced cloud storage. Computers & Security, 74, 299-307.
[14] Fernandes, E., Jung, J., & Prakash, A. (2016). Security analysis of emerging smart home applications. Proceedings of the IEEE Symposium on Security and Privacy (SP), 636-654.
[15] Kothmayr, T., Schmitt, C., Hu, W., Brunig, M., & Carle, G. (2013). DTLS based security and two-way authentication for the Internet of Things. Ad Hoc Networks, 11(8), 2710-2723.
[16] Hammi, M. T., Hammi, B., Bellot, P., & Serhrouchni, A. (2018). Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. Computers & Security, 78, 126-142.
[17] M. Y. Aalsalem, M. Al-Qurishi, M. M. Muwafaq, A. A. Zaidan and M. O. Ahmed, "Towards a blockchain-based secure Internet of Things infrastructure," 2019 International Conference on Intelligent Computing and Its Emerging Applications (ICEA), Riyadh, Saudi Arabia, 2019, pp. 1-6.
[18] R. C. Pereira, A. T. R. Pozza, J. A. P. Lotufo and F. J. Von Zuben, "Blockchain-based access control for the Internet of Things," 2019 IEEE International Smart Cities Conference (ISC2), Casablanca, Morocco, 2019, pp. 383-388.
[19] N. Makhdoom, A. Abolhasan, S. K. Nepal and R. Abbas, "Blockchain's adoption in IoT: The challenges, and a way forward," 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 2019, pp. 460-465.
[20] X. Zhang, P. Fan, L. Chao, X. Tang and X. Yang, "Blockchain-based privacy-preserving authentication for securing IoT applications in smart cities," 2019 IEEE International Conference on Smart Internet of Things (SmartIoT), Tianjin, China, 2019, pp. 1-6.
[21] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, ''Blockchain for IoT security and privacy: The case study of a smart home,'' in Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops), Mar. 2017, pp. 618–623, doi: 10.1109/PERCOMW.2017.7917634.
[22] K.-L. Wright, M. Martinez, U. Chadha, and B. Krishnamachari, ''SmartEdge: A smart contract for edge computing,'' in Proc. IEEE Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData), Jul. 2018, pp. 1685–1690, doi: 10.1109/ Cybermatics_2018.2018.00281.
[23] A. Moinet, B. Darties, and J.-L. Baril, ''Blockchain based trust & authentication for decentralized sensor networks,'' 2017, arXiv:1706.01730.
[24] 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, doi: 10.1007/S10586-020-03058-6.
[25] L. Gong, D. M. Alghazzawi, and L. Cheng, ''BCoT sentry: A blockchain-based identity authentication framework for IoT devices,'' Information, vol. 12, no. 5, pp. 1–20, 2021, doi: 10.3390/ INFO12050203.
[26] M. Zhaofeng, M. Jialin, W. Jihui, and S. Zhiguang, ''Blockchain-based decentralized authentication modeling scheme in edge and IoT environment,'' IEEE Internet Things J., vol. 8, no. 4, pp. 2116–2123, Feb. 2021, doi: 10.1109/JIOT.2020.3037733.
[27] A. Armando, R. Carbone, L. Compagna, J. Cuellar, and L. Tobarra, ''Formal analysis of SAML 2.0 web browser single sign-on: Breaking the SAML-based single sign-on for Google apps,'' in Proc. 6th ACM Workshop Formal Methods Secur. Eng. (FMSE), 2008, pp. 1–9, doi: 10.1145/1456396.1456397.
[28] O. Mir, M. Roland, and R. Mayrhofer, ''DAMFA: Decentralized anonymous multi-factor authentication,'' in Proc. 2nd ACM Int. Symp. Blockchain Secure Crit. Infrastruct., 2020, pp. 10–19, doi: 10.1145/3384943.3409417.
[29] M. T. Hammi, P. Bellot, and A. Serhrouchni, ''BCTrust: A decentralized authentication blockchain-based mechanism,'' in Proc. IEEE Wireless Commun. Netw. Conf. (WCNC), Apr. 2018, pp. 1–6, doi: 10.1109/WCNC.2018.8376948.