Decentralized Smart City of Things: A Blockchain Tokenization-enabled Architecture for Digitization and Authentication of Assets in Smart Cities

Usman Khalil*
School of Digital Science, Universiti Brunei Darussalam,
Jalan Tungku Link, Gadong BE1410, Brunei Darussalam.
E-mail address: 19h8340@ubd.edu.bn
uskhalil@gmail.com

Owais Ahmed Malik
School of Digital Science, Universiti Brunei Darussalam,
Jalan Tungku Link, Gadong BE1410, Brunei
Darussalam.Institute of Applied Data Analytics, Universiti
Brunei Darussalam, Jalan Tungku Link, Gadong BE1410,
Brunei Darussalam.E-mail address:
owais.malik@ubd.edu.bn

Ong Wee Hong
School of Digital Science, Universiti Brunei Darussalam,
Jalan Tungku Link, Gadong BE1410, Brunei Darussalam.
E-mail address:
weehong.ong@ubd.edu.bn

Mueen Uddin (Sr. Member IEEE)
College of Computing and IT, the University of Doha for
Science and Technology, Doha, 24449, Qatar.E-mail
address: mueen.uddin@udst.edu.qa
mueen.malik@ieee.org

ABSTRACT

IoT-enabled smart devices have become an essential part of the smart city architectures, which establishes all the underlying architectures to operate altogether, such as the Internet of Things (IoT), Cyber-Physical Systems (CPSs), Internet of Cyber-Physical Things (IoCPT), and Internet of Everything (IoE). Similarly, these underlying architectures constitute a system to realize the concept of smart cities and, ultimately, a smart planet. A private blockchain-based architecture, Decentralized Smart City of Things (DSCoT), has been proposed in this study which utilizes Blockchain tokenization (i.e., Non-fungible tokens-NFTs) for the representation and authentication of user and IoT assets by defining smart device attributes. Through NFTs, the uniqueness of the IoT assets and users has been realized, which helps digitize these assets as non-interchangeable units of data stored on a digital ledger. The proposed architecture ensures this functionality of unique asset representation by deploying smart contracts and further for IoT assets and user authentication. The mechanism provides security services such as confidentiality, integrity (using SHA-III one-way encryption), availability, and authorization (CIA). The evaluation of the proposed functions and components has been provided in terms of Gas consumption and time complexity, showing promising results. An innovative approach of functions to query the smart contract for the status of assets in the NFT registry offers no transaction cost (in Ether/Gewi), making the proposed extension efficient in terms of time complexity. This architecture aims to provide a smart city solution that may ensure robust security features utilizing Blockchain, NFTs, and SHA-III encryption mechanisms.

CCS CONCEPTS

• **Computer system organization**; • **Architectures**; • **Distributed architectures**; • **Peer-to-peer architectures**; • **Embedded and cyber-physical systems**; • **Sensors and actuators**;

KEYWORDS

Decentralized Ledger Technology, Private Blockchain, Hyperledger Besu, Smart City, Cyber-Physical System, IoT, Blockchain Tokenization, Security Services, Authentication, SHAIII, Smart contract

ACM Reference Format:
Usman Khalil*, Owais Ahmed Malik, Ong Wee Hong, and Mueen Uddin (Sr. Member IEEE). 2022. Decentralized Smart City of Things: A Blockchain Tokenization-enabled Architecture for Digitization and Authentication of Assets in Smart Cities. In *2022 7th International Conference on Cloud Computing and Internet of Things (CCIOT 2022), September 23–25, 2022, Okinawa, Japan.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3569507.3569513

1 INTRODUCTION

A rapid increase in the development and consumption of IoT-enabled smart devices has established an association among the underlying architectures to operate altogether under a smart city architecture. The architectures, such as the Internet of Things (IoT), Cyber-Physical System/s (CPSs), and Internet of Cyber-Physical Things (IoCPT), constitute an architecture of the Internet of Everything (IoE), which provides functionality to the concept of smart cities and, ultimately, a smart planet. The goal of smart city architecture is to achieve robustness for a solution that may integrate all the real-time response applications in terms of people, processes, and data. These things play specific roles and work together to enable future cities and communities to give rise to the concept of smart cities [1]. The devices with this architecture will be connected to the internet to communicate data and information, making industries, healthcare, and cars more intelligent and efficient [2].

Though different researchers have different opinions, smart city architecture can mainly be divided into three-layered architecture,

CCIOT 2022, September 23–25, 2022, Okinawa, Japan
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9673-8/22/09…$15.00
https://doi.org/10.1145/3569507.3569513

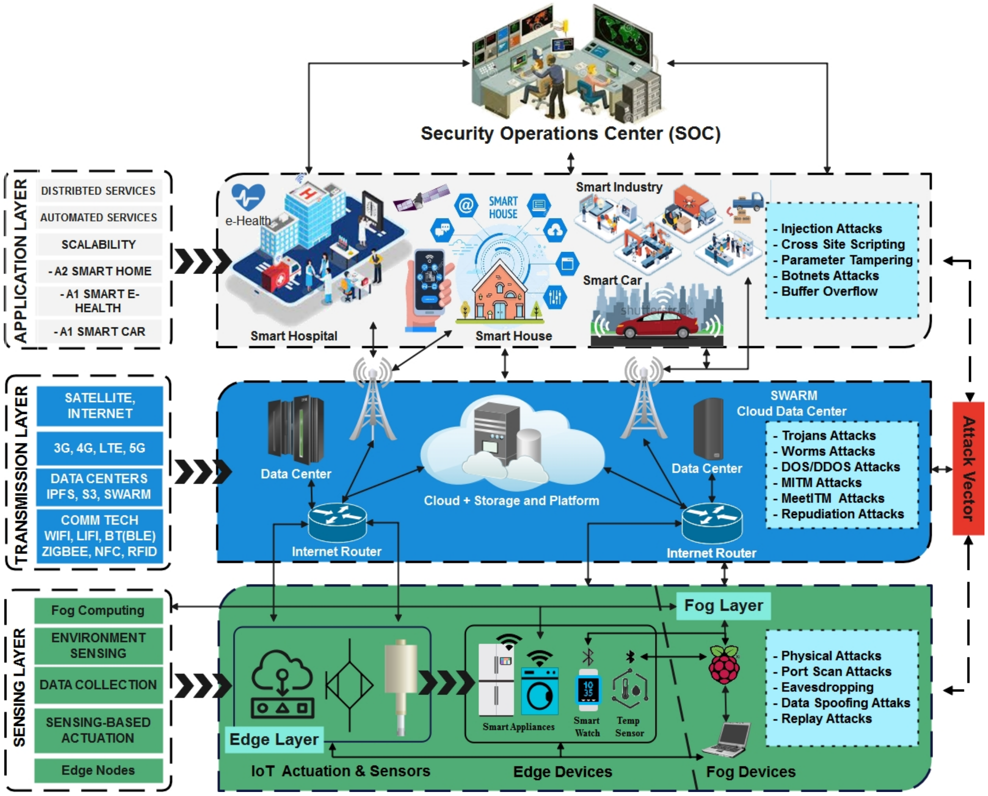


Figure 1: Generalized Smart City Architecture and the Attack Vector

as shown in Figure 1 [3]. The context of CPSs under smart cities has more complex large-scale systems developed and deployed at the industry level, such as the SCADA system (Supervisory Control and Data Acquisition) [1]. Further shown in the figure, the smart city architecture can be organized into layers based on the assets operating in a physical cyberspace environment that provides connectivity with the network for data flow, such as the internet. The data captured by the physical assets, i.e., sensors and actuators, are processed at the physical layer, referred to as the sensing layer. The command-and-control, together with Security Operations Center (SOC), works under the application layer, defines the applications for the asset's behavior at the sensing layer. The network provides connectivity using communication and transmission technologies at the transmission layer. In the smart cities context, these CPSs are managed by the national and private organizations that work in conjunction with government bodies such as the municipal committees. The SOC connects to the internet to deploy the functionality using cloud platforms and services (i.e., cloud services, cloud storage services, and cloud management services) [4]. Smart cities face issues when connected to the internet to enforce automation.

The physical process and the cyber system constitute a CPS. The architecture manages the physical processes, such as a network of resource-constrained devices with sensing and actuation properties [5]. These cyber-physical systems constitute a smart city infrastructure that needs safeguards against adversaries who exploit these systems for personal gains or sabotage system automation. In smart city infrastructure, the data is transmitted from multiple CPSs to the SOC over the internet, posing security threats in different communication architectures of the smart city, as shown in Figure 1. The main concerns are identifying the privacy and authenticity of users and assets, i.e., sensors and actuators, and deploying a consensus mechanism with low latency. Apart from centralized architecture, distributed systems have also been in use traditionally, but the authentication mechanism for the smart city based on decentralized systems has to be explored for its use in distributed ledger technology (DLT). In this context, energy-efficient and low latent consensus mechanisms must be deployed to attain immutability. The representation of assets by a unique identifier as a possession of an owner and virtually defining the IoT assets is a need via smart contracts that may support not modifying the current state of the hardware for device identification.

1.1 The Proposed Decentralized Smart City of Things

The convergence of user and device authentication schemes based on decentralized architectures provides a new dimension to attain robust security yet provokes new challenges. Since the Blockchain-enabled mechanisms have not been limited only to a specific domain (i.e., crypto-currency), the solutions based on the distributed environment have become an obvious choice to attain security. The proposed blockchain-based architecture has been presented in Section 3, which adds a Blockchain (BC) layer to the generalized smart city-layered architecture presented earlier in Figure 1. It integrates IoT-enabled smart devices in blockchain-enabled CPSs (such as smart homes, hospitals, etc.). The proposed blockchain-enabled smart city architecture can be classified into four layers, including the blockchain layer that supports robust security mechanisms. The underlying distributed ledger technology provides decentralization to the proposed mechanism, while the consensus mechanisms provide robust security for communication that cannot be tempered. The posted data is shared among all the nodes in the BC network, making it decentralized and in an immutable state.

Tokenization in BC presents the concept of digital representation of an asset on the Blockchain or colloquially "programmable asset." The non-fungibility of assets, however, is important to represent its ownership which has to be unique to claim the ownership rights. CryptoKitties is one of the first-ever Ethereum-based collectibles game use cases deployed ERC721 Non-Fungible Tokens (NFTs) in a production environment [6]. The NFTs are a more complex version of the ERC20 fungible tokens, having extensions that are split across several smart contracts in different environments. The NFTs represent the ownership of physical or digital assets such as physical property, virtual collectibles, or negative value assets. Although NFTs have been defined under the category of currency tokens, these crypto tokens can be used apart for specified purposes. This property has been used in the proposed architecture to represent the physical assets for IoT device representation and authentication mechanism. Tokens presented by BC tokenization are algorithms implemented as a Smart Contract on a Blockchain which sets the research focus on the points mentioned below in terms of the main contribution.

- We explore and discuss smart city layered architectures for employing authentication schemes and device representation in smart city scenarios keeping in view the underlying architecture.
- We propose an NFT-based protocol for the digital representation and authentication of IoT-enabled smart devices that utilize blockchain architecture for the smart city.
- The proposed protocol focuses on the digital representation and authentication of IoT-enabled smart devices from a software standpoint which does not require additional hardware upgrades from the manufacturer, such as Physical Unclonable Functions (PUF) or on-chip SRAM.
- We developed smart contracts that have been deployed over private Ethereum-based Blockchain (i.e., Hyperledger Besu) for authentication, representation, and to attain security services (i.e., confidentiality, data integrity, availability, authentication, and authorization).

The rest of the paper is organized as follows. Section 2 discusses the literature review, security services, and security issues in smart city architecture. Section 3 presents the methodology of the NFTs-based proposed architecture for the user, fog, and smart devices authentication over Hyperledger Besu and related protocols. Section 4 thoroughly discusses the details of the proposed ERC721 extension. Section 5 reviews the results, and finally, a concise conclusion is presented at the end.

2 LITERATURE SURVEY

Several surveys discuss the security challenges posed to IoT-enabled smart assets in a smart city context [7]. The literature survey has been carried out considering security and asset representation aspects, i.e., representation and authentication mechanism for admin/owner, users, and IoT assets in a distributed IoT architecture for the smart city. Recently, blockchain-based architectures have been proposed to represent assets and components; however, the assets utilizing *NFTs* for security and asset representation in the literature are explicitly lacking. We intend to imply and propose the NFT functionality based on the literature reviewed.

2.1 IoT-enabled Smart Device Representation

IoT assets embedded with physically integrated chips (ICs) have been utilized to represent smart devices to mitigate the exploitation of smart assets from intangible and physical adversaries, which poses a limitation in modifying the current state of the hardware. This property provides a physically defined "digital fingerprint" as a unique identifier. Based on the distributed architectures, the authors in [8] proposed a blockchain-based platform solution for IoT device authentication, data privacy, and security service via smart contracts. The proposed mechanism uses a defined function on the IC named Physically Unclonable Functions (*PUFs*), which implies the authentication mechanism factors. The IoT device hardware was tailored to meet blockchain performance.

The authors in [9] also exploit embedded ICs utilizing PUF with blockchain tokenization to represent assets by a unique identifier as possessing an owner. The authors proposed a smart Non-fungible token (smartNFT) that is physically bound to its IoT device. This mechanism also defines authentication mechanisms based on Physical Unclonable Functions (*PUFs*), which describe the physical properties of the devices and are used to identify and represent the devices using their private key and BCA address. This smartNFT can establish secure communication channels with owners and users and operate dynamically with several modes associated with their token states.

2.2 Blockchain-based Mechanism

The authors in [10] proposed authentication and access control mechanisms based on a distributed architecture for lightweight IoT devices. The Elliptic Curve Digital Signature Algorithm (*ECDSA*) has been used for key generation, generating public and private keys for the devices and the fog nodes.

A proposed framework in [11] *BCoT Sentry* (Blockchain of Things Sentry) integrates Blockchain with an IoT network. The authors present a novel approach to the feature selection method (similar feature selection method in machine learning utilizing the maximum information coefficient (MIC), used to measure the discrimination of IoT devices). The smart contract defines the device's identity information and related operations and is triggered once the transactions in the Blockchain are posted.

A blockchain-based decentralized authentication modeling scheme named *BlockAuth* has been proposed in [12]. The authentication scheme was claimed to be suitable for password-based, certificate-based, biotechnology-based, and token-based authentication for high-level security requirement systems in Edge and IoT

Table 1: Evaluation of State-of-the-Art Authentication Mechanisms based on Blockchain

Proposed Mechanism	Blockchain Platform	Consensus Mechanism	Mutual Auth	Access Control	Data Integrity	Data Anonymity
Blockchain-based IoT Authentication, 2021 [8]	Ethereum H/ledger Fabric	PoW PBFT	✓	×	✓	×
smartNFT-based PUF Mech, 2021 [9]	Ethereum	PoW	✓	×	✓	×
Blockchain-based Authentication System, 2020 [10]	Ethereum	PoW	✓	✓	✓	×
BCoT Sentry, 2021 [11]	Ethereum	PoW	×	×	✓	✓
BlockAuth, 2021 [12]	H/ledger Fabric 1.4	PBFT	✓	×	✓	×
SmartEdge- Ethereum, 2018 [13]	Ethereum	PoW	✓	×	✓	×
DAMFA, 2020 [14]	Namecoin	PoW	✓	×	✓	✓
BCTrust, 2018 [15]	Ethereum	PoW	✓	✓	✓	×
Blockchain-based User Authentication, 2018 [16]	Ethereum	PoW	✓	×	✓	×

environments. A blockchain-based decentralized authentication protocol has been developed using the Blockchain's consensus and smart contract capability.

An Ethereum-based smart contract for edge computing has been proposed as *SmartEdge* in [13] for its low-cost, low-overhead tool for compute-resource management. The authors showed the design breakdown of a smart contract into three key steps and described them in the context of their design of *SmartEdge*. The performance was evaluated in terms of low-overhead delay in executing a job and transaction costs (Ether/Gwei) that should not be significant relative to the crypto value.

Alternate to SSO (*Single Sign-On*) for a one-time password authentication scheme, the authors in [14] proposed a new Distributed Anonymous Multi-Factor Authentication (*DAMFA*) scheme that uses public Blockchain (i.e., Bitcoin & Namecoin). The underlying consensus mechanism improves usability, which builds on a Threshold Oblivious Pseudorandom Function (*TOPRF*) for resistance to offline attacks. It requires no interaction with the identity provider; hence, the user's authentication no longer depends on a trusted third party.

A framework *BCTrust* for the authentication mechanism based on Blockchain has been proposed in [15]. It has been designed especially for devices with resource constraints such as computational, storage, and energy consumption constraints. The robustness claimed by the authors is because of the underlying framework of the public Blockchain, and the smart contracts provide access control over authentication mechanisms for system (SID) and User or Device identification (UID).

Blockchain-enabled fog nodes for user authentication schemes have been proposed in [16], which deploy smart contracts to authenticate users to access IoT devices. A distributed system based on the public blockchain design has been proposed with its implementation using Ethereum smart contracts for IoT device authentication at scale.

2.3 Problems Associated with the State-of-the-Art (SOTA) Authentication Mechanisms

While many works in the application of Blockchain in IoT systems have been reported in the literature, each has addressed one or few security aspects in authentication, access control, data integrity, and data anonymity. However, the proposed architecture has been tested on the private BC platform Hyperledger Besu which deploys IBFT 2.0, an energy-efficient and low latent consensus mechanism. The exploitation of NFTs to represent assets by a unique identifier as a possession of an owner and to virtually define the IoT assets in a proposed ERC721 extension via smart contracts supports not modifying the current state of the hardware for device identification, as presented in [9]. It gives the mechanism low latency in terms of time complexity, while the solution has been observed to have energy-efficient Gas consumption. Table 1 summarizes the achievement of the state-of-the-art Blockchain-based authentication mechanisms. We further elaborate on the issues in the following sections.

2.3.1 Distributed Platforms and Consensus Mechanism Issues. Most proposed mechanisms have been deployed on the Ethereum platform, utilizing the traditional Proof of Work (PoW) consensus mechanism. Ethereum is undoubtedly a platform that supports public, private, and hybrid blockchains to be developed and deployed; it also provides the option to utilize decentralized applications (dApps) to provide logic to execute the functions as required. However, the consensus mechanism poses performance issues of fault tolerance, decentralization, stability, and high-level security.

- Other platforms, such as Hyperledger Besu [17], Hyperledger Fabric [18], Solana [19], etc., provide much more efficient consensus mechanisms for developing solutions over smart contracts.
- These platforms support more energy-efficient and low latent consensus mechanisms such as IBFT, IBFT 2.0, and Clique. These consensus mechanisms must imply the robust fault tolerance, decentralization, stability, and high-level

security and authentication stability of IoT-enabled smart devices to support the smart city infrastructure.

2.3.2 Assets Digitization Issues. Enhancement to the representation of IoT-enabled smart devices has not been sufficiently studied. This property will help the devices increase security from the device's abuse in the case of adversaries. The devices in the existing works have been represented through traditional media access control (*MAC)* and internet protocol (*IP)* addresses in a network.

- Recently, Physically Unclonable Functions (PUFs), as discussed in Section 2.1, have been the choice and low-cost solution to identify devices for solutions implemented on the Blockchain.
- In a recent study [9], NFTs have been utilized to represent assets by a unique identifier as a possession of an owner, but these tokens were employed to bound the IoT assets physically employing PUFs.
- The asset representation also defines authentication mechanisms based on PUF, which describe the physical properties of the devices and are used to identify and represent the devices using their private key and Blockchain address.
- Since the ERC721 extension proposed in [9] is hardware-dependent, it requires a hardware upgrade from the manufacturer, which may incur manufacturing costs.
- The binding of NFT with the hardware properties may fail the overall system in case of device malfunction.
- With hardware upgrades, the IoT assets have been noticed to have increased initialization time, which incurs latency issues such as initializing Bootloader, located in the main SoC's internal OTP memory.
- The coding of the Bootloader cannot be modified since it is the device's Root of Trust (RoT).
- Hence, the on-chip SRAM (Static Random-Access Memory), also considered an SRAM PUF, cannot be altered and poses time complexity, computational complexity, and latency issues.

2.3.3 Smart Contract Issues. Smart contracts (SC) define applications that are decentralized in nature and are special entities that provide real-world data in a trusted manner.

- Functions and events in the SCs enable the actuation mechanisms to be employed in the IoT-enabled smart devices much faster. Still, every transaction has a cost in Ether/Gwei and consumes more Gas which is inefficient for an IoT use case.
- Smart contract deployment with defined authentication functions may provide security, but transaction costs may result in latency as functions implying authentication have been computationally expensive. Hence, efficient and lightweight authentication schemes with lower computational costs are a need to fill the research gap.

2.3.4 Security Issues – Manufacturer's Perspective. On the other hand, IoT-enabled smart devices have security issues from the manufacturer's perspective, as the asset's firmware is not fully equipped with a security mechanism by default.

- Especially authentication, access control schemes, and firmware updates are commonly found unattended, posing these assets' exploitation.
- New strong and lightweight encryption schemes such as SHAIII would help mitigate the authentication and access control issues based on communication and computational costs.
- The state-of-the-art authentication schemes are computationally expensive as they mostly rely on the functions and events rather than developing functions to save transaction costs and Gas consumption.

3 METHODOLOGY

The methodology is based on proposing an extension of the ERC721 standard as a Decentralized Smart City of Things (The proposed architecture) for smart cities. Figure 2 depicts the proposed architecture that represents the Owner, users, fog, and IoT-enabled smart devices and authenticates the assets by utilizing the proposed ERC721 extension in respective CPSs. It utilizes newly defined attributes for digitizing the assets from a software standpoint, omitting the need to update the assets' hardware. The proposed architecture proposes an extension for deploying smart device representation through non-fungible tokens, NFT-based Externally Owned Accounts (EOAs), and their authentication mechanism via smart contracts. The implementation of a private distributed technology blockchain has been carried out, as indicated in Figure 2.

The decentralized application (dApp) functions at the application layer by deploying smart contracts. These smart contracts are stored in the NFT registry in blockchain storage and engage as required by respective CPSs, including smart hospitals, smart homes, smart industry, smart cars, etc. It also ensures functionality by deploying smart contracts to authenticate the admin/owner and remote users at the application layer and authenticating the fog and assets at the sensing layer, as shown in Figure 2. The proposed architecture provides security in terms of confidentiality and availability by utilizing constructors and modifiers, limiting the access control to the admin/owner to manage admins and access functions execution. The data integrity and anonymity have been provided utilizing the SHA-III family one-way encryption mechanism. The proposed architecture blockchain layer deploys the Hyperledger Besu, enabling the proposed architecture's distributed functionality with a robust consensus mechanism. NFT-based EOAs in this layer has also represented these components. NFTs are unique and non-interchangeable units of data stored on a distributed ledger. Thus, utilizing blockchain tokenization in the proposed infrastructure without a centralized third-party intervention will provide mechanisms to digitally define the assets and components and attain a robust authentication mechanism.

As shown in Figure 2, the assets initialization triggers if the proof of ownership is verified (msg.sender). The Owner is the creator/admin of the smart contract, and only the Owner can add, delete or map the fog devices with the IoT assets providing confidentiality and availability to all assets and components in the proposed architecture. The transaction (Tx) info, details of NFT-based EOAs of the User, fog, IoT assets, proposed metadata, and TokenID are stored in the NFT registry to authenticate the assets accordingly. Further shown in the figure, the authentication layer adds the authentication and authorization mechanisms that deploy a decentralized application to provide the authentication logic for

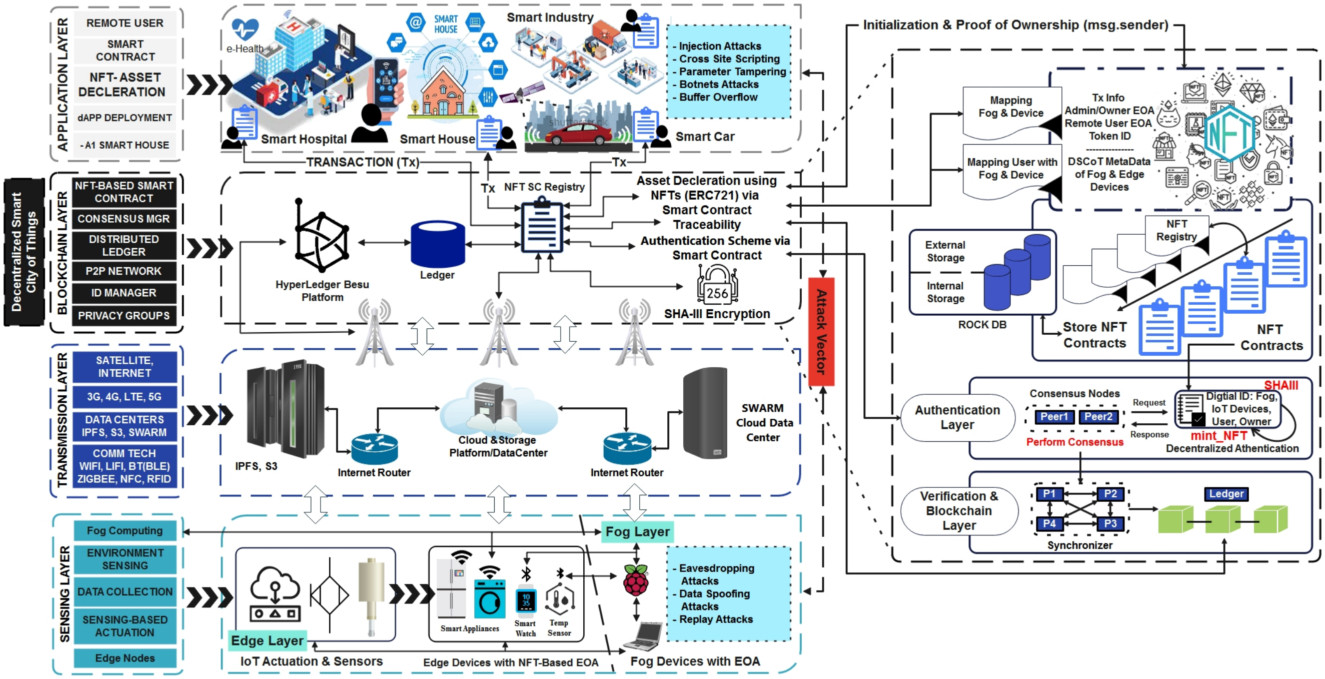


Figure 2: The proposed Decentralized Smart City of Things

the connected nodes in CPSs in a smart city context. Specifically, the proposed architecture helps integrate robust asset authentication, exploiting the functionality of the SHAIII encryption protocol. It has been deployed in the "*mintNFT*" function at the blockchain layer in proposed mechanism. Its additional uses for the function, such as an authenticated encryption system, leverages faster hashing in the proposed architecture. Since centralized systems, including key management systems, may jeopardize the system's security because of trusted third-party service providers, the cryptosystems based on decentralized technology have been opted to enable the solutions deployed on top of the blockchain solutions.

Once user authentication with fog and IoT assets completes, the consensus mechanism triggers, which in this case is IBFT 2.0, and the transaction is posted to the peers in the P2P network, a group of synchronizer nodes. It synchronizes before being posted to the blockchain ledger as an immutable transaction. The posted transactions would provide traceability as the unique identifying codes of an NFT, and the digitization of every asset can easily be traced down in the distributed ledger.

4 PROPOSED IOT-ENABLED ERC721-IOT EXTENSION

Smart contracts help develop a client-side application that runs on top of the Blockchain as a decentralized app (dApp) [20]. These applications are developed in Solidity. Remix IDE (v0.23.3) has been used to develop, compile and deploy the smart contracts for the proposed architecture based on the ERC721 standard. An NFT is a unique and non-interchangeable unit of data stored on a digital ledger (Blockchain). The NFTs are built on the Ethereum Request for Comment ERC-721 [21], which defines a standard interface using wallet applications to work with any NFT on Ethereum platforms, i.e., Hyperledger Besu.

The smart devices with token$_{\mathrm{id}}$ and NFT-based Externally Owned Accounts (EOAs) are referred to as resource owners [22], [23]. ERC-721, in contrast to its predecessor, the ERC-20 (fungible and interchangeable) tokens, are non-interchangeable and have uniqueness for each assigned asset. This lucrative property makes its use for Smart devices distinction (non-fungible). NFTs provide two basic attributes for identifying the uniqueness of assets, i.e., token identification (token$_{\mathrm{id}}$) and NFT-based EOA that can be owned, transferred, and approved to act on their behalf. However, smart devices require additional attributes to represent the devices at fog and edge layers, with functions defining their functionality. Hence, an extension of the ERC-721 standard is presented for smart devices since more attributes would be helpful for asset representation and the authentication mechanism to validate the users' and assets' authenticity, as depicted in Table 2.

The proposed metadata of the standard attributes defined in the ERC721 standard, such as Owner (address) and token ID, are utilized in DSCoT to validate the Owner/Admin, which means only the Owner can access the smart contract for managing the resources, and execute changes through functions. The U$_{\mathrm{ID}}$ attribute was created to represent the user identification which represents the user who can access the devices mapped to the respective fog devices, and DID represents the device identification. The Owner/Admin can only provide access control to the users for accessing the assets. Similarly, FogID represents the fog node identification while T and ΔT represent the block timestamp and change the time for the blockstamp to record the replay or spoofing attacks, respectively.

Table 2: Proposed Extension Metadata

Sr #	Attribute	Description
1	Owner/Admin	EOA of the Owner
2	tokenId	Token ID of the Owner
3	$\mathrm{U_{ID}}$	User Identification
4	$\mathrm{D_{ID}}$	Smart Device Identification
5	$\mathrm{Fog_{ID}}$	Fog Device Identification
6	T	Time Stamp
7	ΔT	Change in Time duration

The description of the attributes has been defined and shown in the table for better understanding.

4.1 Components and Functions of Proposed ERC721-IoT Extension

The smart contract based on the proposed ERC721 extension has been designed to expand its functionality to different CPSs in smart city architecture. Hence the designed components can be integrated as required, such as smart homes, smart hospitals, smart supply chains, etc. The main components of the extension in the smart contract are the Owner (admin), the user, IoT-enabled smart device, and the fog device. As depicted in Table 3, the components with functions and events are defined in the interfaces, while the main functions were developed in The proposed Smart Contract. The ERC721 was imported using the OpenZeppelin Contracts, which provide flexibility regarding combining these as useful custom extensions [24]. The proposed ERC721 extension was developed in five categories to realize the smart city concept.

As depicted in Table 3, the first category defines functions and events to add, delete, and check the number of admins via NFT EOAs. The *approve()* function for ERC721 was utilized to approve other admins with the given token identification (ID) for updating and calling the proposed functions. Once mapped, the transaction is posted on distributed ledger where privacy is maintained until the resource owner updates or removes the details. Since the extension has been designed to fulfill the smart city IoT-based architecture, the transfer is out of the scope of this research; we want to call the contract only by the owner token or an approved operator to preserve the data privacy and integrity in the smart city architecture. Here NFT tokenization plays an important role in representing the smart devices (IoT & fog) utilizing EOAs generated with NFT token identifications (IDs). These IDs are unique for each device and other stakeholders in the proposed extension, such as admin/owner and users.

The second category defines functions and events to add, delete, and map the fog and IoT devices. The fog node in respective CPSs needs to have the functionality of mapping the NFT-based EOAs of the IoT assets. To map the devices, the Owner initiates the *DeviceFogMapping()* function to map the fog node with the respective IoT assets, as depicted in Table 3. Once the fog node maps with IoT devices using EOAs, the devices can be assigned to the user who can access these devices as and when required. After mapping, the transaction is posted on the distributed ledger, where the data integrity is maintained until the resource owner updates or removes the details.

The third category in Table 3 defines functions and events to add, delete, and map the Users, IoT, and fog devices. After *DeviceFogMapping()*, the fog nodes are mapped to the respective IoT assets; the user can be assigned by its NFT-based EOA, which will be mapped to the fog node by initializing the *UserDeviceMapping()* function. It would assign the user to the fog node with its mapped IoT assets. Once the user is mapped to its fog node and IoT assets, it will be ready to proceed with NFT minting to initiate the authentication process.

The fourth category defines the ownership functionality, while the balance of the component EOAs can be verified in the fourth step, using ERC721 *balanceOf* and *ownerOf* operators, as shown in Table 3. The functionality has been adopted from the ERC721 standard to check the balance of the Owner and User for all the transaction costs (in Ether/Gewi). Since every transaction has to be performed over BC, it will be helpful to make function calls to check the balance (Ether/Gwei) and the EOA_{Owner} and EOA_{User} for transaction-making purposes, respectively. It will also be helpful to manage the devices through functions as mentioned in the 1st, 2nd, 3rd and final steps of the proposed mechanism.

The final category defines functions and events for the NFT minting, which receives the EOAs after the user is allocated to the nodes in the third step, i.e., *UserDeviceMapping(*EOA_{User}, EOA_{Fog}, EOA_{Device}*)* to apply the authentication aforementioned. It is the final step where the authentication process will authenticate the assets once the NFT-based EOAs of the Users, fog, and IoT devices are mapped with each other or will reject otherwise. The process generates the NFT $\mathrm{Token_{Id}}$ for the user using the SHA-III algorithm. The generated NFT $\mathrm{Token_{Id}}$ will be a unique identification code used for user authentication whenever the user wants to access the devices. All these functions emit the events that mention the operator details, such as EOAs with $\mathrm{token_{ids}}$ for admin, user, IoT, and fog devices.

5 RESULTS AND DISCUSSION

After the testbed was deployed, the methods imposed by NFTs using smart contracts need to be validated by the amount of Gas consumed in carrying out the transactions on the personalized Besu platform. The Ethereum blockchain platform uses the cryptocurrency ether (ETH), while the smaller fractions are measured in Gwei.

Table 3: Proposed ERC721-IoT Extension Components & Functions

Functions	Metadata

Functions & Events to add/Del and check the No. of Admins/EOAs	function approve(address _approved, uint256 _tokenId) external payable;
	event AdminAdded(address indexed newAdmin, address indexed addingAdmin);
	event AdminAlreadyExists(address indexed newAdmin, address indexed sender);
	function No_ofAdmins() external view returns (uint256);
	function adminAdd() external view returns (address[] memory);
	function delAdmin (address admin) external;
	event AdminDeleted(address indexed newAdmin, address indexed deletingAdmin);

Functions & Events to Add/Del/Map devices (IoT, Fog)	function DeviceFogMapping(address fog, address device) external;
	event FogDeviceMappingAdded(address indexed fog, address indexed device, address indexed addingAdmin);
	event FogDeviceAllMappingDeleted(address indexed fog, address indexed deletingAdmin);
	event DeviceDoesnotExist(address indexed device, address indexed fog, address indexed sender);
	function delDev(address fog) external;

Functions & Events to add/Del/Map Users with Smart devices	function UserDeviceMapping(address user, address device, address fog) external;
	event UserDeviceAllMappingDeleted(address indexed user, address indexed deletingAdmin);
	event UserDeviceMappingAdded(address indexed user, address indexed device, address addingAdmin, address indexed fog);
	function delUser(address user) external;

Functions & Events to check balance and Owner of token.	function balanceOf(address _owner) external view returns (uint256);
	function ownerOf(uint256 _tokenId) external view returns (address);

Minting Functions & Events for User and devices Authentication Mechanism	function mintNFT(address device, address fog) external;
	event Authenticated(address indexed user, address indexed device, address indexed fog);
	event NotAuthenticated(address indexed user);
	event InvalidUser(address indexed device, address indexed fog, address indexed sender);
	event TokenCreated(bytes32 indexed _tokenID, address indexed user, address device, address indexed fog, uint256 timestamp);

Gas is the execution result of the operation that needs to modify the data on the Blockchain. The decentralized app (dApp) execution, such as a smart contract, spends Gas to allocate resources defined. A lightweight decentralized app implementation would cost a lesser Gas limit, which means less work to execute a transaction using ETH or a smart contract. More Gas would be consumed, resulting in an inefficient solution. The proposed functions evaluation at the time of deployment was carried out for the gas consumption so that the cost of each function may be known. Figure 3 shows the Gas consumed by the main functions, where the mint function has consumed more Gas which is expected for the encryption and authentication of users and devices. *UserDeviceMapping* the user to the respective fog and IoT nodes have also consumed more Gas. In contrast, the rest of the functions, such as *approve(), delAdmin(), delDev(),* and *DeviceFogMapping()* functions, have consumed almost the same amount of Gas on average. In contrast, the *delDev()* function to delete the allocated devices was observed to have consumed the lowest amount of Gas.

A comparison of the efficiency of the proposed solution in terms of Gas consumption has been made with [9], which depicts efficient gas consumption for the proposed architecture than smartNFT-based PUF's main minting functions. The proposed *mintNFT()* vs. smartNFT-based PUF *createToken()* was observed to be approximately ≈ 27% while a proposed *approve()* vs. smartNFT-based PUF *startOwnerEngagement()* was observed to be approximately ≈ 11% more efficient respectively.

5.1 Time Complexity

The proposed extension does not modify data on the Blockchain to verify the identity of all the functions and components. An innovative approach with the functions has been designed to query the smart contract for the status of assets in the NFT registry. It would not amend any data on the chain but will help save the transaction cost (Ether/Gewi) and be efficient in terms of time complexity, as shown in Figure 4.

- The "adminAdd()" call() method has been designed to find the admins/Owners addresses, as shown in Figure 4, which shows "0x5B38Da6a701c568545dCfcB03FcB875f56beddC4" as an admin NFT-based EOA.

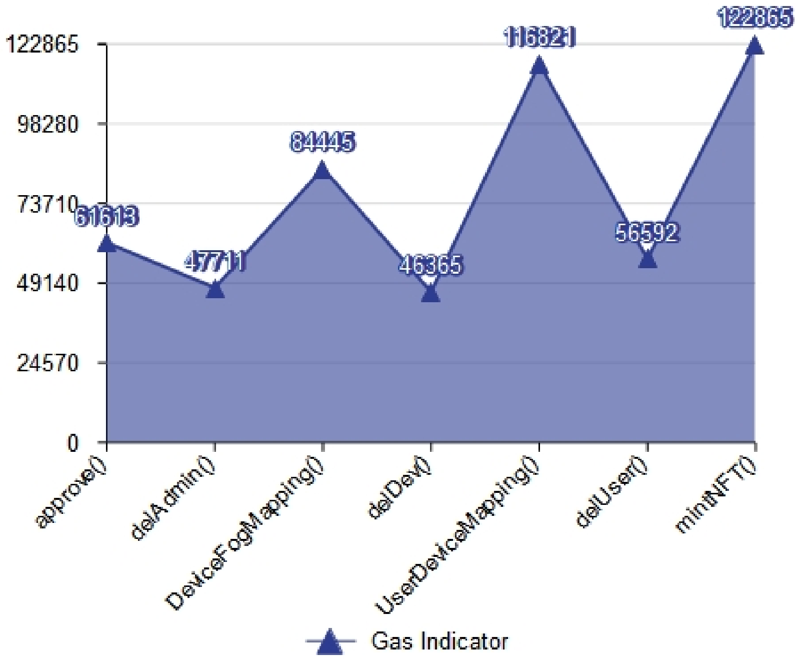
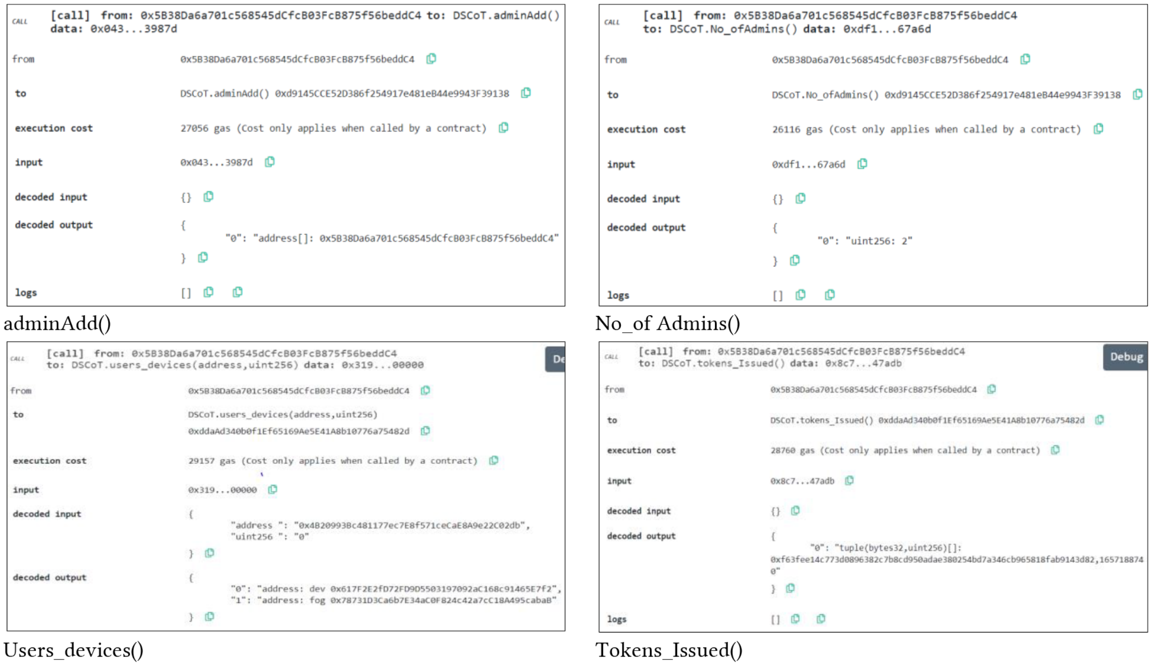


Figure 3: Decentralized Smart City of Things Functions Gas Indicator



```
CALL    [call]  from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: DSCoT.adminAdd()
        data: 0x043...3987d

from              0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

to                DSCoT.adminAdd() 0xd9145CCE52D386f254917e481eB44e9943F39138

execution cost    27056 gas (Cost only applies when called by a contract)

input             0x043...3987d

decoded input     {}

decoded output    {
                          "0": "address[]: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4"
                  }

logs              []
```

adminAdd()

```
CALL    [call]  from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
        to: DSCoT.No_ofAdmins() data: 0xdf1...67a6d

from              0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

to                DSCoT.No_ofAdmins() 0xd9145CCE52D386f254917e481eB44e9943F39138

execution cost    26116 gas (Cost only applies when called by a contract)

input             0xdf1...67a6d

decoded input     {}

decoded output    {
                          "0": "uint256: 2"
                  }

logs              []
```

No_of Admins()

```
CALL    [call]  from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
        to: DSCoT.users_devices(address,uint256) data: 0x319...00000

from              0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

to                DSCoT.users_devices(address,uint256)
                  0xddaAd340b0f1Ef65169Ae5E41A8b10776a75482d

execution cost    29157 gas (Cost only applies when called by a contract)

input             0x319...00000

decoded input     {
                          "address ": "0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db",
                          "uint256 ": "0"
                  }

decoded output    {
                          "0": "address: dev 0x617F2E2fD72FD9D5503197092aC168c91465E7f2",
                          "1": "address: fog 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB"
                  }
```

Users_devices()

```
CALL    [call]  from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4
        to: DSCoT.tokens_Issued() data: 0x8c7...47adb

from              0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

to                DSCoT.tokens_Issued() 0xddaAd340b0f1Ef65169Ae5E41A8b10776a75482d

execution cost    28760 gas (Cost only applies when called by a contract)

input             0x8c7...47adb

decoded input     {}

decoded output    {
                          "0": "tuple(bytes32,uint256)[]:
                  0xf63fee14c773d0896382c7b8cd950adae380254bd7a346cb965818fab9143d82,165718874
                  0"
                  }

logs              []
```

Tokens_Issued()

Figure 4: Time Complexity of Proposed ERC721-IoT Extension

- The “No_ofAdmins()” call() method has been designed to find the total number of admins/Owners, as shown in Figure 4, which shows “2” admin addresses exist.
- The “user_Devices_Add()” call() method has been designed to find the total number of devices mapped to a specific user, as shown in Figure 4, which shows that two NFT-based EOAs exist, i.e., fog:

"0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB" and IoT device:

- "0x617F2E2fD72FD9D5503197092aC168c91465E7f2"
- The "tokens_Issued()" call() method has been designed to find a total number of NFTs, as shown in Figure 4, which shows generated NFT with a block timestamp.
- "0xf63fee14c773d0896382c7b8cd950adae380254bd7a346cb965818fab9143d82, 1657188740".

Hyperledger Besu is an Ethereum-based private chain in which the time to generate new blocks depends on the block size. The transactions that cost low transaction fees or computational complexity can have delays, but increasing transaction fees can solve the problem. We made more than 500 calls to the functions mentioned above, found it efficient, and did not find the transaction charging any gas fees. Assuming that there are 'n' IoT devices that require identity authentication, the proposed architecture presents O(n) time complexity.

6 CONCLUSION

An extension of non-fungible tokens (NFTs) based on the ERC-721 standard for digitizing assets and utilizing these digital tokens for the authentication mechanism of these assets in a smart city architecture has been proposed. An NFT-based smart contract has been developed using IDE on a private blockchain utilizing a robust consensus mechanism, i.e., IBFT 2.0. At the same time, the implementation was carried out for all the functions and procedures using NFT-based EOAs assigned to all the components in the smart contract. It has been successfully deployed, providing security services such as Confidentiality, Integrity, Availability (CIA), and Authorization. Since NFTs are non-interchangeable and unique to represent each physical asset's ownership, a mechanism has been devised for user and device authentication. In contrast, smart devices have also achieved digital representation (IoT, fog) utilizing the NFT ERC721 standard. The evaluation of the proposed functions and components has been carried out in terms of Gas consumption and time complexity, showing promising results. An innovative approach of functions has been designed to query the smart contract for the status of assets in the NFT registry. It does not amend any data on the chain, thus helping save the transaction cost (in Ether/Gewi) and making the proposed extension efficient in terms of time complexity. The use case scenarios for smart houses and smart hospitals will be deployed in the future, and results will be reported accordingly.

ACKNOWLEDGMENTS

The author would like to express sincere appreciation for the resources provided by Universiti Brunei Darussalam (UBD), Brunei Darussalam.

Declarations

CONFLICT OF INTEREST

The author(s) declared no potential conflicts of interest concerning this article's research, authorship, and/or publication.

REFERENCES

[1] S. Mitchell, N. Villa, M. Stewart-weeks, and A. Lange, "The Internet of Everything for Cities the ' Livability ' of Cities and Communities Cities: Fertile Ground for Realizing IoE Value," 2013.

[2] U. Khalil, Mueen-Uddin, O. A. Malik, and S. Hussain, "A Blockchain Footprint for Authentication of IoT-Enabled Smart Devices in Smart Cities: State-of-the-Art Advancements, Challenges and Future Research Directions," *IEEE Access*, vol. 10, pp. 76805–76823, 2022.

[3] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Sensing as a service model for smart cities supported by Internet of Things," *Trans. Emerg. Telecommun. Technol.*, vol. 25, no. 1, pp. 81–93, 2014.

[4] J. P. A. Yaacoub, O. Salman, H. N. Noura, N. Kaaniche, A. Chehab, and M. Malli, "Cyber-physical systems security: Limitations, issues and future trends," *Microprocess. Microsyst.*, vol. 77, p. 103201, 2020.

[5] E. K. Wang, Y. Ye, X. Xu, S. M. Yiu, L. C. K. Hui, and K. P. Chow, "Security issues and challenges for cyber physical system," in *Proceedings - 2010 IEEE/ACM International Conference on Green Computing and Communications, GreenCom 2010, 2010 IEEE/ACM International Conference on Cyber, Physical and Social Computing, CPSCom 2010*, 2010, pp. 733–738.

[6] J. Castro *et al.*, "CryptoKitties | Collect and breed digital cats!," 2020. [Online]. Available: https://www.cryptokitties.co/. [Accessed: 27-May-2022].

[7] U. Khalil, O. A. Malik, M. Uddin, and C.-L. Chen, "A Comparative Analysis on Blockchain and Centralized Authentication Architectures for IoT-Enabled Smart Devices in Smart Cities: A Comprehensive Review , Recent Advances , and Future Research Directions," *Sensors*, vol. 22, no. 12, pp. 1–52, 2022.

[8] M. N. Islam, "Enabling IoT Authenticaiton, Privacy and Security via Blockchain," 2021.

[9] J. Arcenegui, R. Arjona, R. Román, and I. Baturone, "Secure combination of iot and blockchain by physically binding iot devices to smart non-fungible tokens using pufs," *Sensors*, vol. 21, no. 9, 2021.

[10] 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, 2020.

[11] L. Gong, D. M. Alghazzawi, and L. Cheng, "Bcot sentry: A blockchain-based identity authentication framework for iot devices," *Inf.*, vol. 12, no. 5, pp. 1–20, 2021.

[12] 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, 2021.

[13] K. L. Wright, M. Martinez, U. Chadha, and B. Krishnamachari, "SmartEdge: A Smart Contract for Edge Computing," in *Proceedings - IEEE 2018 International Congress on Cybermatics: 2018 IEEE Conferences on Internet of Things, Green Computing and Communications, Cyber, Physical and Social Computing, Smart Data, Blockchain, Computer and Information Technology, iThings/Gree*, 2018, pp. 1685–1690.

[14] O. Mir, M. Roland, and R. Mayrhofer, "DAMFA: Decentralized anonymous multi-factor authentication," *BSCI 2020 - Proc. 2nd ACM Int. Symp. Blockchain Secur. Crit. Infrastructure, Co-located with AsiaCCS 2020*, pp. 10–19, 2020.

[15] M. T. Hammi, P. Bellot, and A. Serhrouchni, "BCTrust: A decentralized authentication blockchain-based mechanism," in *IEEE Wireless Communications and Networking Conference, WCNC*, 2018, vol. 2018-April, pp. 1–6.

[16] R. Almadhoun, M. Kadadha, M. Alhemeiri, M. Alshehhi, and K. Salah, "A User Authentication Scheme of IoT Devices using Blockchain-Enabled Fog Nodes," in *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA*, 2019, vol. 2018-Novem.

[17] Hyperledger Foundation, "Hyperledger Besu," 2021. [Online]. Available: https://limechain.tech/blog/hyperledger-besu-explained/. [Accessed: 08-Jan-2022].

[18] H. H. Pajooh, M. Rashid, F. Alam, and S. Demidenko, "Hyperledger fabric blockchain for securing the edge internet of things," *Sensors (Switzerland)*, vol. 21, no. 2, pp. 1–29, 2021.

[19] Solana, "Scalable Blockchain Infrastructure: Billions of transactions & counting | Solana: Build crypto apps that scale," 2022. [Online]. Available: https://solana.com/. [Accessed: 05-Jun-2022].

[20] Remix ©, "Remix - Ethereum IDE." 2019.

[21] W. Entriken, "ERC-721." [Online]. Available: http://erc721.org/. [Accessed: 25-May-2022].

[22] T. Moonstream, "An analysis of 7 , 020 , 950 NFT transactions on the Ethereum blockchain," 2021.

[23] N. S. William Entriken, Dieter Shirley, Jacob Evans, "EIP-721: Non-Fungible Token Standard," *Ethereum Improvement Proposals*, 2018. [Online]. Available: https://eips.ethereum.org/EIPS/eip-721. [Accessed: 21-Sep-2021].

[24] OpenZeppelin, "ERC721 - OpenZeppelin Docs," 2022. [Online]. Available: https://docs.openzeppelin.com/contracts/4.x/erc721. [Accessed: 27-May-2022].