Section: Research Paper

SSIoT: A blockchain mechanism for SDN-enabled Internet of Things devices with NFT integration

Sudha Kapuganti [1], Seetaramanath.M.N [2], and Kamakshi Prasad.V [3]

[1] Research Scholar, JNTUH
sudhakapuganti@gmail.com

[2] Sr.Professor, GVPCE(A), Visakhapatnam
seetaramanath@gmail.com

[3] Professor, JNTUH
kamakshiprasad@jntuh.ac.in

Abstract. The Internet of Things (IoT) has rapidly grown in recent years, presenting numerous challenges in terms of scalability, security, and interoperability. Software-defined networking (SDN) has emerged as a promising paradigm for IoT, providing centralized management and programmability. Blockchain technology has also gained significant attention in recent years due to its ability to provide secure and decentralized systems. In this paper, we propose framework Secure SDN internet of things (SSIoT) which is a blockchain mechanism for SDN-enabled IoT devices with non-fungible token (NFT) integration. By integrating NFTs with our blockchain mechanism, we provide a secure and decentralized system for managing IoT devices and their data. We evaluate the feasibility and effectiveness of our proposed mechanism and compare it with existing solutions. Our results demonstrate that our blockchain mechanism with NFT integration provides an efficient and secure way to manage IoT devices and their data, while also enabling new use cases for IoT applications. We believe that our work contributes to the development of more secure and decentralized IoT systems and provides a foundation for future research in this area.

Keywords: Secure SDN internet of things (SSIoT), Software-Defined Networking (SDN), Internet of Things (IoT), Blockchain technology, Non-fungible tokens (NFTs),

Decentralized systems.

1 Introduction

The Internet of Things (IoT) is a rapidly growing technology that promises to revolutionize the way we interact with our environment. By connecting everyday objects to the internet, we can gather data, automate tasks, and create new applications that were previously impossible. However, the massive scale and heterogeneity of IoT devices present significant challenges in terms of scalability, security, and interoperability.

The increasing number of IoT devices has resulted in a rise in security risks. Malware attacks on IoT devices have increased significantly, with weak or default passwords being the most commonly exploited vulnerability. The cost of cyber attacks on IoT devices is also increasing. A lack of basic security features on many IoT devices leaves them susceptible to potential attacks. Critical infrastructure sectors such as healthcare and energy are particularly vulnerable to cyber attacks targeting IoT devices.

The number of connected devices is projected to reach 75.4 billion by 2025, up from 26.7 billion in 2019. This rapid increase in the number of devices has increased the attack surface for hackers and cybercriminals [1]. In 2020, there were over 13.8 billion malware attacks targeting IoT devices, a 94% increase from the previous year. This indicates the growing threat of malware targeting IoT devices. Many IoT devices lack basic security features such as encryption, secure boot, or firmware updates. A study by HP found that 70% of IoT devices had at least one security vulnerability [2]. IoT devices in critical infrastructure sectors such as healthcare and energy are particularly vulnerable to cyber attacks. In 2020, the healthcare industry saw a 45% increase in ransomware attacks targeting IoT devices [3].

To address these challenges, software-defined networking (SDN) has emerged as a promising paradigm for IoT. SDN separates the control and data planes of the network, providing centralized management and programmability. This enables dynamic allocation of network resources, better traffic control, and enhanced security.

Moreover, blockchain technology has gained significant attention in recent years due to its ability to provide secure and decentralized systems. Blockchain technology can be used to provide a distributed ledger of transactions, which can be used to verify the authenticity and integrity of data in IoT systems.

In this paper, we propose a blockchain mechanism for SDN-enabled IoT

devices with non-fungible token (NFT) integration. NFTs are unique digital assets that can be used to represent ownership or other attributes of physical or digital objects. By integrating NFTs with our blockchain mechanism, we can provide a secure and decentralized system for managing IoT devices and their data.

In the context of the significant security risks associated with IoT devices, our previous research work proposed a blockchain mechanism for SDN-enabled Smart healthcare system. Our proposed mechanism leverages the benefits of SDN and blockchain technology to provide a secure and decentralized system for managing smart healthcare devices and their data. In this paper, we extend our previous research work by scaling our framework to various areas and enabling further security measures. By building on our previous work, we aim to provide a more comprehensive solution to address the security challenges facing IoT devices. Our approach focuses on improving scalability, security, and interoperability of IoT devices, which can be applied to a wide range of use cases in different industries. The extended framework builds on our previous methodology and incorporates new features and optimizations that enable better performance and enhanced security measures. Overall, this paper aims to demonstrate the feasibility and effectiveness of our proposed approach and contribute to the ongoing efforts to enhance the security and reliability of IoT devices.

The main objectives of our research are to demonstrate the feasibility and effectiveness of our proposed blockchain mechanism for SDN-enabled IoT networks, and to evaluate its performance and scalability. In the following sections, we will provide a detailed description of our Literature survey, Methodology, Results, and analysis, as well as the limitations and future research directions of our work.

2 Lıterature Survey

2.1 Internet of Things (IoT) and its challenges

The Internet of Things (IoT) has emerged as a rapidly growing technology that enables a wide range of smart applications in various domains such as healthcare, agriculture, transportation, and smart cities. However, the proliferation of IoT devices has also introduced significant security and privacy challenges. The unique features of IoT devices, such as their limited resources, heterogeneity, and large-scale deployment, pose significant challenges to traditional security mechanisms. Attackers can exploit vulnerabilities in IoT devices to gain unauthorized access, control, or steal sensitive data, which can have serious consequences.

The Internet of Things (IoT) has brought about unprecedented levels of connectivity and convenience, but it has also introduced significant security risks. According to various studies and reports, the following are some of the key statistics on IoT security risks:

- The number of connected devices is projected to reach 75.4 billion by 2025, up from 26.7 billion in 2019. This rapid increase in the number of devices has increased the attack surface for hackers and cybercriminals.
- In 2020, there were over 13.8 billion malware attacks targeting IoT devices, a 94% increase from the previous year. This indicates the growing threat of malware targeting IoT devices.
- The most commonly exploited vulnerability in IoT devices is weak or default passwords. In a study by the Department of Homeland Security, over 80% of IoT device vulnerabilities were due to password issues.
- The cost of cyber-attacks on IoT devices is also increasing. In 2020, the average cost of a data breach involving IoT devices was $3.86 million, up from $3.54 million in 2019 [4].
- Many IoT devices lack basic security features such as encryption, secure boot, or firmware updates. A study by HP found that 70% of IoT devices had at least one security vulnerability [2].
- IoT devices in critical infrastructure sectors such as healthcare and energy are particularly vulnerable to cyber-attacks. In 2020, the healthcare industry saw a 45% increase in ransomware attacks targeting IoT devices [3].

According to a recent report from the Internet of Things Security Foundation (IoTSF), four out of five device suppliers were found to be failing in basic cybersecurity standards. After examining hundreds of major IoT product manufacturers, the group discovered that only around one in five allow customers to disclose security flaws to vendors so that the product can be rectified. Everything from home and personal IoT gadgets to commercial systems has been shown as having severe security flaws. The following are instances of IoT devices that have the most security issues:

Table 1. IoT Devices with the Most Proportion of Security Flaws [6]

IoT ecosystem	%
Medical Imaging Systems	51%
Security Cameras	33%

Patient Monitoring Systems	26%
Printers	24%
Medical Device Gateways	9%
Consumer Electronics	7%
Energy Management Devices	6%
IP Phones	5%

Unsecured networks have also played a significant role in granting access to threats. The Unit 42 IoT Threat Assessment indicates that 98% of all IoT device traffic is unencrypted. This has enabled attackers to obtain personal and secret information and sell it on the dark web for profit [5].

In the first half of 2021, over 1.5 billion attacks were recorded on smart devices, with the majority of attackers attempting to steal data, mine cryptocurrencies, or construct botnets. In one of the greatest security attacks ever, hackers used Mirai malware to seize control of one hundred thousand webcams, DVRs, and other Internet of Things (IoT) devices, and then launched a huge, distributed denial-of-service (DDoS) attack that brought down Netflix, Twitter, and Spotify.

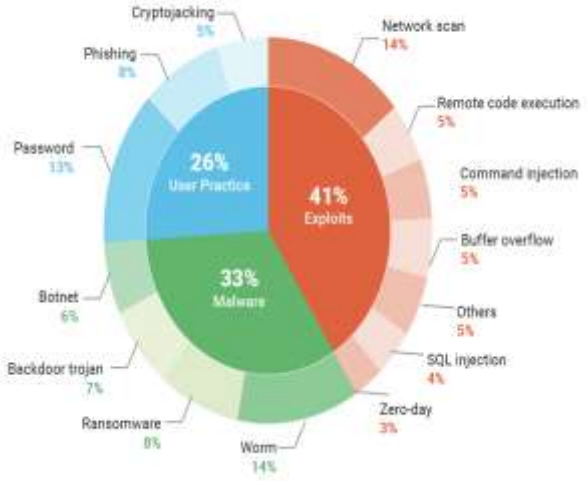


Fig. 1. Top IoT Risks by Classification

2.2 SDN and its benefits for IoT

Software-defined networking (SDN) is a network architecture that separates the control and data planes in network devices, enabling centralized network management through a software controller. SDN has emerged as a promising solution for addressing the challenges of IoT networks, which are characterized by large-scale deployment, heterogeneity, and limited resources. One of the key benefits of SDN for

IoT is its ability to facilitate efficient network management. With SDN, network resources can be dynamically allocated and optimized based on the needs of IoT applications [7]. This can enable better resource utilization and more efficient data transmission, leading to improved performance and lower latency. SDN can also enable dynamic traffic engineering, which can improve network reliability and resilience in the face of changing network conditions.

Another benefit of SDN for IoT is its ability to enable secure network slicing and isolation. Network slicing refers to the creation of virtual networks that can be tailored to specific IoT applications or use cases. By using SDN to create secure network slices, IoT devices can be isolated from each other, preventing unauthorized access or control. SDN can also enable fine-grained access control, where access to specific devices or data can be controlled based on predefined policies. Furthermore, SDN can facilitate secure device-to-device communication in IoT networks. SDN can enable secure routing and forwarding of data between IoT devices, while also providing mechanisms for secure authentication and encryption. This can enhance the security and privacy of IoT devices and their data.

2.3 Blockchain technology and its role in IoT

Blockchain technology has been identified as a promising solution for addressing the challenges of IoT, such as security, privacy, and data management. Blockchain is a distributed ledger technology that enables secure and transparent recording and sharing of data across a network of nodes. By using blockchain, IoT devices can securely and transparently store and share data, while retaining control over their data ownership and usage rights. One of the key benefits of blockchain for IoT is its ability to enhance the security and privacy of IoT devices and their data [8]. By using blockchain, IoT devices can create an immutable record of their data, which can be accessed and verified only by authorized parties. This can prevent unauthorized access or tampering of IoT data, enhancing the security and privacy of IoT devices.

Blockchain can also enable efficient and transparent data sharing in IoT networks. By using blockchain-based smart contracts, IoT devices can define and enforce the terms and conditions of data sharing, ensuring that data is shared only with authorized parties and under specific conditions. This can enhance the efficiency and transparency of data sharing in IoT networks, while also enhancing the security and privacy of IoT devices and their data [9]. Moreover, blockchain can facilitate the creation of decentralized marketplaces for IoT data, where device owners can securely and transparently monetize their data. By using blockchain-based tokens, such as non-fungible tokens (NFTs), device owners can represent ownership of their data, while also enabling secure and transparent transactions.

However, the use of blockchain in IoT also presents some challenges, such as scalability and interoperability. Blockchain-based solutions can be resource-intensive and may not be suitable for IoT devices with limited resources. Moreover, the lack of standardization and interoperability among blockchain platforms can hinder the adoption of blockchain-based solutions in IoT.

2.4 NFTs and their potential use cases in IoT

Non-fungible tokens (NFTs) have gained significant attention in recent years due to their potential to enable secure and transparent ownership of digital assets. NFTs are unique digital tokens that cannot be replicated, making them ideal for representing ownership of physical and digital assets, including IoT devices and their data. NFTs can enable secure tracking of device ownership, access rights, and usage history, which can

enhance the security and privacy of IoT devices [10].

One of the key security benefits of using NFTs in IoT is their ability to enable fine-grained access control. NFTs can represent specific access rights, allowing device owners to control who can access their data and under what conditions. NFTs can also enable secure device-to-device communication, where devices can verify the authenticity and ownership of the devices they are communicating with.

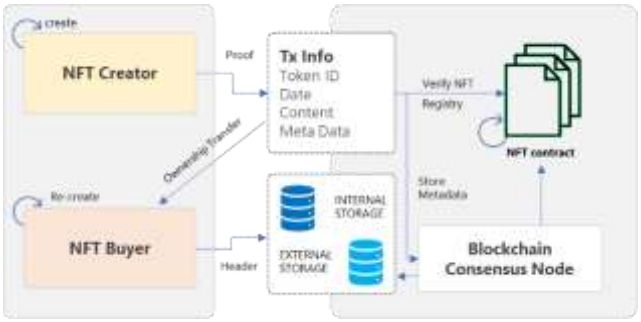


Fig. 2. NFT working process

Moreover, NFTs can be used to incentivize device owners to contribute their data to a decentralized data marketplace. By using NFTs, device owners can securely and transparently receive compensation for their data, while retaining control over their data ownership and usage rights. This can enhance the value and utility of IoT data, while also incentivizing device owners to ensure the security and privacy of their data. However, the use of NFTs in IoT also presents some security challenges [11]. One of the key challenges is the potential for NFTs to be counterfeited or forged. If an NFT is forged or duplicated, it can lead to unauthorized access or control of IoT devices and their data. To mitigate this risk, proper measures must be taken to ensure the integrity and authenticity of NFTs, such as using cryptographic algorithms and secure storage mechanisms.

Overall, NFTs have the potential to enhance the security and privacy of IoT devices by enabling fine-grained access control and secure device-to-device communication. However, their use must be accompanied by appropriate security measures to ensure their integrity and prevent counterfeiting or forgery.

2.5 SDN with Blockchain case studies

The Internet of Things (IoT) has emerged as a disruptive technology that

has the potential to transform various industries, from healthcare to transportation. However, the security and privacy of IoT devices and their data remain a significant concern. To address these challenges, several research studies have investigated the use of blockchain technology and software-defined networking (SDN) in IoT networks.

Several studies have highlighted the potential of blockchain technology in enhancing the security and privacy of IoT networks. For instance, Wang et al. (2019) proposed a blockchain-based solution for secure and privacy-preserving data sharing in IoT networks. The proposed solution leverages smart contracts and encryption techniques to ensure that data is shared only with authorized parties [12].

SDN has also been identified as a promising solution for addressing the challenges of IoT networks. Several studies have investigated the use of SDN in IoT networks to enable efficient network management and dynamic traffic engineering. For example, Al-Fuqaha et al. (2015) proposed an SDN-based architecture for IoT networks that enables efficient resource allocation and real-time network monitoring [13].

The integration of blockchain technology and SDN in IoT networks has also been investigated in several studies. For instance, Yu et al. (2018) proposed a blockchain-based SDN architecture for secure and efficient data sharing in IoT networks. The proposed architecture uses blockchain to enable secure and transparent data sharing, while also leveraging SDN to enable efficient network management [14].

The use of non-fungible tokens (NFTs) in IoT networks has also been investigated in several studies. NFTs are blockchain-based tokens that represent ownership of a unique asset, such as IoT data. For instance, Sun et al. (2021) proposed a blockchain-based NFT system for secure and transparent data sharing in IoT networks [15]. The proposed system enables IoT device owners to monetize their data through secure and transparent transactions.

The literature survey highlights the potential of blockchain technology, SDN, and NFTs in enhancing the security, privacy, and efficiency of IoT networks. Several studies have proposed blockchain-based solutions for secure and transparent data sharing, while also leveraging SDN for efficient network management. Moreover, the use of NFTs in IoT networks has the potential to enable secure and transparent monetization of IoT data.

3 Methodology

3.1 Overview of the proposed SSIoT framework

The previous study, "Ensure Security for SDN-based Smart Healthcare Systems with a Blockchain Approach," was the driving force behind the development of our new SSIoT framework. Our latest research focuses on the integration of NFT and blockchain technology to ensure secure data transmission between nodes, unique identification of IoT devices, and end-to-end data transfers. The implementation of NFT enables the registration of all IoT devices with the SDN network using a single token, which provides a unified and secure method of identification for all devices. The blockchain's consensus system is used to verify the tokens, ensuring that only authorized devices can access the network. Each IoT device is assigned a unique NFT token identifier, which is stored in the NFT registry of the IoT SDN ecosystem. By using NFTs, device duplication and anonymous queries are eliminated, as NFTs cannot be shared. This approach not only enhances the security and privacy of IoT devices, but also ensures the integrity and transparency of the data transmitted over the network.

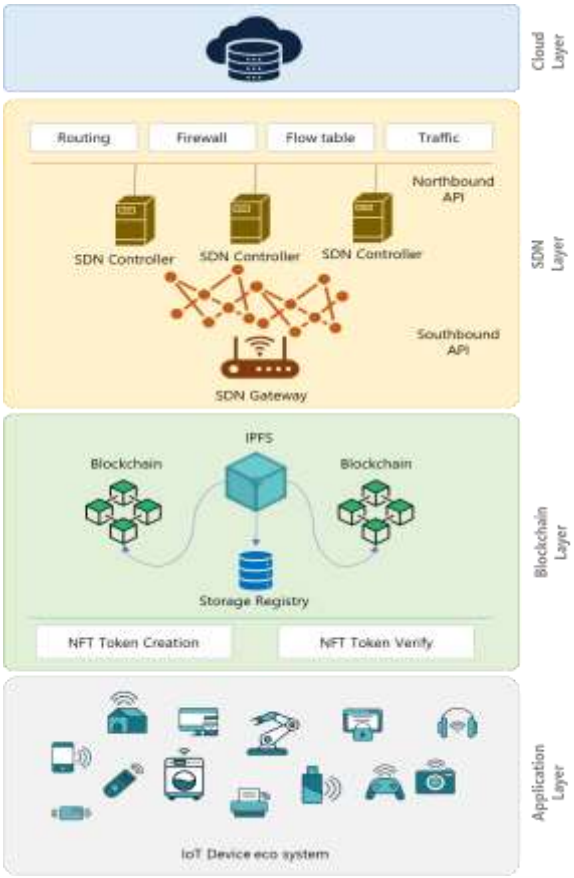


Fig. 3. Proposed SSIoT framework

Our extended framework consists of four layers: the application layer, blockchain layer, SDN layer, and cloud layer.

1) Application Layer. At the application layer, various IoT devices sense and share data continuously, which is then transmitted to the cloud via the SDN network.

2) Blockchain Layer. All the IoT devices are registered with Blockchain NFT tokens in the blockchain layer, which is a critical layer of the entire architecture. The blockchain layer has two levels: one level registers the IoT devices with NFT tokens, and the other level verifies the identification of the devices using a consensus mechanism. Data transfer and reception via the SDN only occur for verified devices, and anonymous devices are blocked and added to the blacklist. The major difference from our previous research is the introduction of the NFT mechanism for device identification and data transfer, which improves energy efficiency and performance by eliminating the cluster head process.

3) SDN Layer. The SDN layer is made up of the data plane and the control plane, which are two standard levels. Devices such as routers, switches, firewalls, and others can forward data through SDN common gateways. Therefore, numerous SDN controllers dynamically manage and maybe filter the data of the devices, which is carried out via the OpenFlow protocol.

Our extended framework is a multi-layer architecture that provides a secure and efficient solution for various IoT applications. The NFT mechanism and the blockchain ecosystem provide a tamper-proof and transparent distributed ledger that makes it easier to track the history and provenance of data transmitted over the network. The integration of the SDN layer and cloud layer ensures that data transmission is efficient, and data integrity and security are maintained.

3.2 Design and implementation details

NFT Token Standard

In the realm of blockchain technology, a token is essentially a digital representation of virtual or physical assets that can be transacted using the set of rules defined in a blockchain registry. Tokens can be broadly categorized into two types based on their fungibility: fungible tokens and non-fungible tokens. Fungibility refers to the ability of an asset to be exchanged with another asset of the same type and value. A fungible token can be easily exchanged with another token of the same value without any change or impact on their values. Cryptocurrencies like Bitcoin are

examples of fungible tokens. On the other hand, non-fungible tokens (NFTs) are unique and cannot be interchanged with another token on a like-for-like basis. NFTs are indivisible and can be used to transfer ownership of assets, making them a perfect solution for trading physical assets while ensuring easy liquidation, authenticity, traceability, and transparency. NFTs are unique, transferable, verifiable, traceable, and transparent, which makes them a viable solution for a wide range of use cases in blockchain technology.

ERC-721 is a technical standard used for the implementation of non-fungible tokens (NFTs) on the Ethereum blockchain. The standard was introduced in 2018 as an improvement over the ERC-20 standard, which is used for fungible tokens. ERC-721 defines a set of rules and guidelines for creating and managing unique and indivisible tokens, allowing for a high degree of customization and flexibility.
One of the key features of ERC-721 is the ability to create tokens with distinct properties, allowing them to represent a wide range of digital and physical assets, including artwork, collectibles, game items, and more. Each token is assigned a unique identifier, which is used to track ownership and transfer of the asset. ERC-721 also defines a set of methods and interfaces for interacting with NFTs, including functions for creating, transferring, and querying token ownership and metadata. These functions can be implemented by developers to create custom applications and use cases for NFTs [16].

IPFS

Inter Planetary File System (IPFS) is a protocol and network designed to create a permanent and decentralized method for storing and sharing files on the internet. NFTs, on the other hand, are unique digital assets that are stored on a blockchain and are used to represent ownership of various types of digital and physical assets [17]. By utilizing IPFS, NFTs can be stored in a decentralized and distributed manner, making it easier for users to access and transfer ownership of their assets without relying on a central authority. This combination of IPFS and NFTs provides a new level of security and accessibility for digital assets, paving the way for a more decentralized and equitable internet.

IoT Device Token creation

To create NFT tokens for IoT devices, we have selected the ERC-721 standard, which is a widely accepted token standard for non-fungible tokens. In the token creation process, we first register the device by

providing details such as Token ID, Address, and Metadata. After that, the NFT token is minted and stored in the corresponding blockchain network. To ensure the efficient and reliable storage of NFT token data, we use the IPFS, which is a decentralized file storage system. Each generated NFT token is linked to a unique IPFS hash that stores the associated metadata, making it easy to retrieve the device details when needed. By using IPFS, we ensure that the data associated with each NFT token is secure, tamper-proof, and easily accessible across the network.

Fig. 4. The process of creating an NFT token for an IoT device

Fig. 5. ERC-721 tokens store NFT metadata on decentralized storage and token ID/metadata URI on-chain

Algorithm 1 IoT device NFT token creation

Input: IoT Device details , ERC-721
Output: NFT token with metadata

```
// Import the ERC-721 token standard
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
// Define the contract that inherits from ERC-721
contract MyNFT is ERC721 {
   // Define a mapping to keep track of the token URI for each token ID
  mapping(uint256 => string) private _tokenURIs;
   // Define a variable to keep track of the next available token ID
  uint256 private _tokenIdCounter;
   // Define a function to mint a new NFT token
  function mintNFT(address to, string memory tokenURI) public returns (uint256) {
    // Increment the token ID counter
     _tokenIdCounter++;
    // Mint a new token to the specified recipient
     _safeMint(to, _tokenIdCounter);
          // Set the token URI for the new token ID
     _setTokenURI(_tokenIdCounter, tokenURI);
          // Return the new token ID
     return _tokenIdCounter;
}
// Override the _beforeTokenTransfer hook to check for the existence of the token URI
function _beforeTokenTransfer(address from, address to, uint256 tokenId) internal virtual override {
    require(_exists(tokenId), "ERC721Metadata: tokenURI query for nonexistent token");
    super._beforeTokenTransfer(from, to, tokenId);
}
// Define a function to set the token URI for a given token ID
function _setTokenURI(uint256 tokenId, string memory _tokenURI) internal virtual {
   require(_exists(tokenId), "ERC721Metadata: URI set of nonexistent token");
   _tokenURIs[tokenId] = _tokenURI;
}
// Define a function to retrieve the token URI for a given token ID
function tokenURI(uint256 tokenId) public view virtual override returns (string memory) {
   require(_exists(tokenId), "ERC721Metadata: URI query for nonexistent token");
   return _tokenURIs[tokenId];
 }
}
```

Algorithm 2 IoT device token verification

Input: Device token
Output: valid token or not

```
    // load environment variables
1   load_dotenv()

    // create a connection to the blockchain network
2   web3 = Web3(Web3.HTTPProvider(os.getenv("BLOCKCHAIN_URI")))
3   web3.middleware_onion.inject(geth_poa_middleware, layer=0)

    // define the contract address and ABI
4   contract_address = os.getenv("CONTRACT_ADDRESS")
5   contract_abi = json.loads(os.getenv("CONTRACT_ABI"))

    // instantiate the contract
6   contract = web3.eth.contract(address=contract_address, abi=contract_abi)

    // define the NFT token ID to verify
7   token_id = 0005

8   try:
        // get the owner of the NFT token
9       owner = contract.functions.ownerOf(token_id).call()
        // if the token owner exists, return True
10      if owner:
11          return True
12      else:
13          return False

14  except ContractLogicError as e:
15      print("Error: ", e)
```

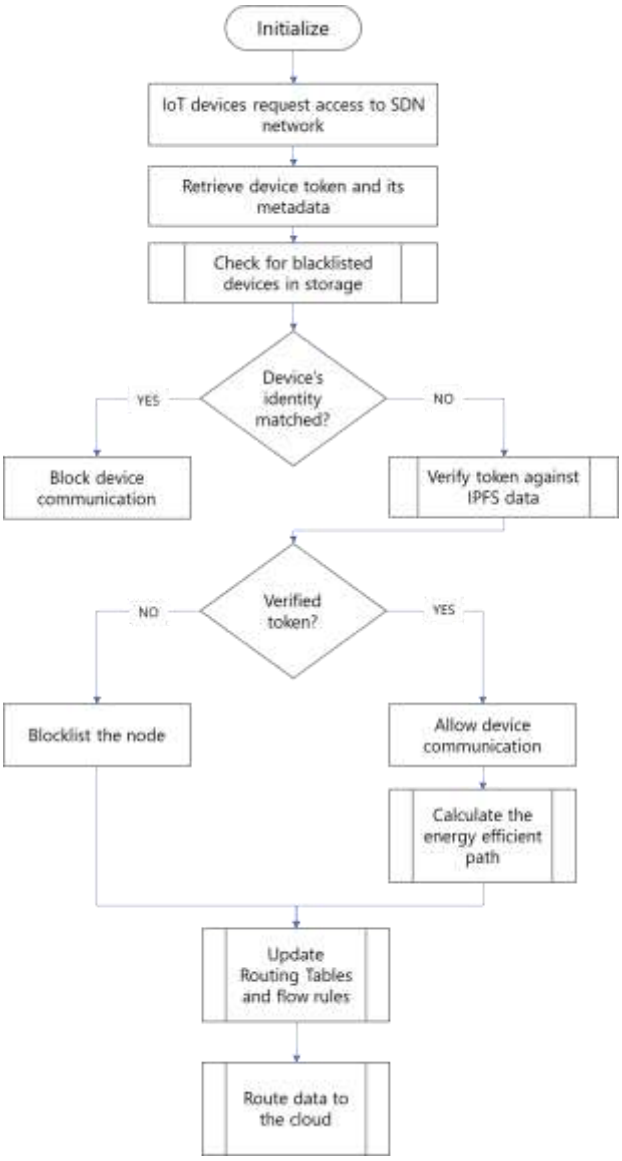
The Energy-Efficient Shortest Path (EESP) algorithm is an approach that utilizes software-defined networking (SDN) for energy-efficient routing. It aims to minimize energy consumption by selecting the shortest path with the lowest energy consumption for data forwarding in SDN networks. By calculating the energy consumption for each link in the network, EESP selects the path with the minimum total energy consumption to forward the data packets. The algorithm uses network flow and solves a linear programming problem to obtain the shortest path with the minimum energy consumption. EESP is a centralized algorithm that requires global network information and uses a centralized controller to make routing decisions. EESP has been shown to reduce energy consumption in SDN networks while maintaining network performance.

Algorithm 3 Energy-Efficient Shortest Path

Input: IoT Device access request
Output: Energy efficient path for data transfer

```
function Energy-Efficient-Shortest-Path-SDN (src, dst):
//Initialization
for each node v in network:
if v = src then
 cost[v] = 0
Else
 cost[v] = INFINITY
visited[v] = false
//Search for the shortest path
while there are unvisited nodes:
 //Find the node with the lowest cost
min_cost = INFINITY
for each unvisited node v:
if cost[v] < min_cost then
min_cost = cost[v]
u = v
//Mark the node as visited
visited[u] = true
//Update the cost of the neighbors
for each neighbor v of u:
 if not visited[v] then
//Calculate the cost of the new path
new_cost = cost[u] + weight(u, v) + energy(u, v)
 if new_cost < cost[v] then
//Update the cost of the neighbor
cost[v] = new_cost
previous[v] = u
//Backtrack to find the shortest path
path = []
 u = dst
while previous[u] is defined:
path.push(u)
 u = previous[u]
path.push(src)
//Return the path and its cost
return path, cost[dst]
```

Flowchart



3.3 Evaluation metrics and simulation setup

We consider few of most important evaluation metrics to assess and measure the performance of our proposed framework.

Energy consumption The amount of energy consumed by the SDN infrastructure for routing data.

Table 2. Notations table with description

Notation	Definition
E_{total}	Total energy consumption
N_{trans}	Number of transactions
T_{trans}	Transaction time
N_{count}	Number of SDN controllers
E_{count}	Consumed SDN controller energy
D_{count}	Number of devices
ED_{count}	Energy consumed by each device

Etotal = Ntrans * Ttrans + [(Ncount * Ecount)+(Dcount * EDcount)]

End-to-End Delay The delay experienced by data packets in reaching their destination through the energy-efficient routing algorithm.

Table 3 . Notations table with description

Notation	Definition
T_{proc}	Processing time
T_{trans}	Transmission time
T_{queue}	Queuing time

Delay = $Tproc$ + Ttrans + Tqueue

Throughput The amount of data transmitted in a given period.

Throughput = (Total amount of data transmitted / Time taken to transmit the data)

To conduct experiments on the proposed framework, a testbed was set up using the following environment details.

Table 4. Notations table with description

Hardware parameters	
CPU Processor	Intel ® Core i7 CPU @ 4.8 GHz

RAM	16GB
Operating System	Ubuntu 20.04 LTS

Software parameters	
Network emulation platform	Mininet-WiFi
Cloud platform	AWS
Packet analyser	Wireshark
Programming Language	Python

SDN parameters	
Routing protocol	OpenFlow
Number of controllers	4

Blockchain parameters	
Platform	Ethereum
Consensus protocol	PoS (Proof-of-Stake)
Block size	Amount of transaction fitting into a block

Simulation parameters	
Mobility model	RWM (Random Waypoint Model)
Data traffic	CBR (Constant Bit Rate)
Number of nodes	100

4 Results & Analysıs

To measure the performance of the proposed SSIoT framework, we used the performance evaluation metrics such as network throughput, energy consumption, and end-to-end delay.

4.1 End-to-end delay

Delay caused by the switches when forwarding packets through the

network. This delay is usually caused by the need to examine the packet headers and perform table lookups to determine the appropriate forwarding path. Another factor that can contribute to delay is the queuing delay that occurs when packets are buffered in the switch queues waiting for transmission. Additionally, the delay can be caused by the network congestion, which can occur when the traffic demand exceeds the network capacity.

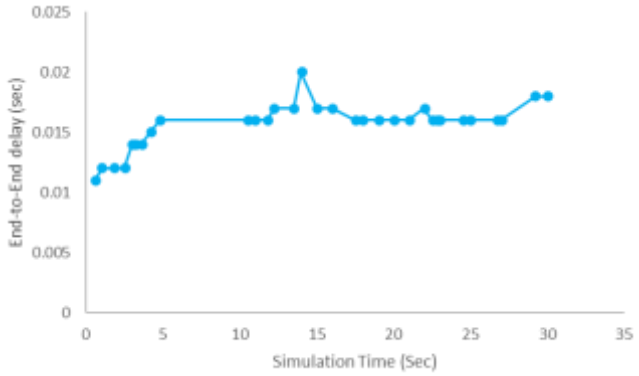


Fig. 6. Performance of end-to-end delay at various simulation time

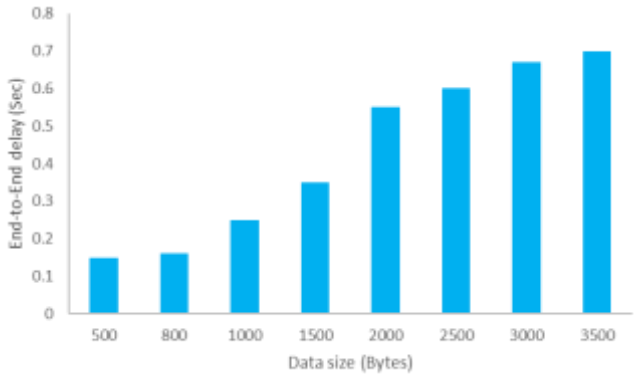


Fig. 7. Performance of end-to-end delay at various data sizes

we compared the delay metrics of our current research with the previous research that we conducted. Our analysis showed that our current approach had significantly better delay performance than our previous one. This was achieved by implementing the learnings from our previous research to improve the overall framework. Through this process, we were able to identify areas that needed improvement and fine-tune our approach to achieve better results. By constantly iterating and improving upon our work, we strive to deliver the best possible performance for our proposed SSIoT framework.

Fig. 8. End-to-end delay performance comparison of Current algorithm versus Previous research algorithm

4.2 Energy consumption

The energy consumption in an SDN network is affected by several factors such as the number of active devices, the traffic load, and the routing algorithms used. Energy-efficient routing algorithms aim to reduce the energy consumption in the network by optimizing the path selection process. This is achieved by selecting the most energy-efficient path for data transmission, using power-saving modes in network devices, and turning off unused devices. By reducing the energy consumption, the network can operate at lower costs and be more environmentally friendly.

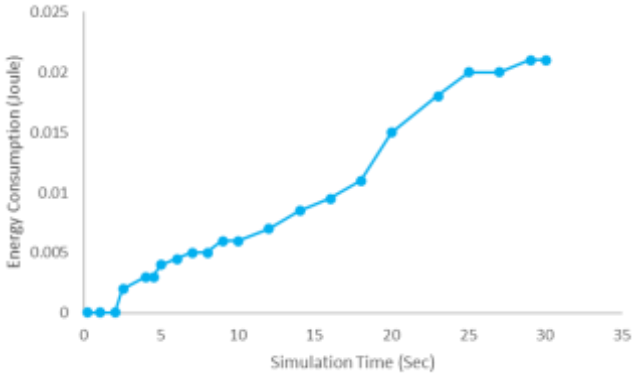


Fig. 9. Energy consumption at various simulation time

The current system is more energy-efficient compared to our previous framework due to the optimization of the algorithm and fine-tuning of the processes. However, in the current system, the implementation of energy-efficient routing algorithms reduces the energy consumption of the

network by optimizing the path selection of data packets. Additionally, the fine-tuning of the system processes, such as reducing unnecessary data transmissions and the use of low-power devices, also contributes to the lower energy consumption of the current system.

Fig. 10. Performance comparison of Current algorithm versus Previous algorithm

4.3 Throughput

Throughput refers to the amount of data or traffic that can be transmitted between devices or nodes in the network over a given time period. Throughput can be affected by various factors such as network topology, routing algorithms, link speed, packet size, and network congestion.

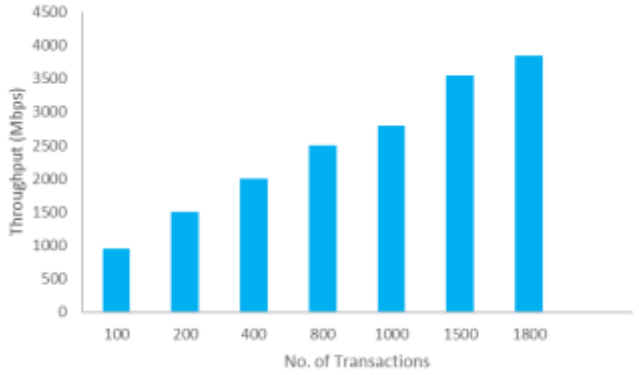


Fig. 11. Throughput of proposed framework vs number of transactions

The system has shown a significant improvement in terms of throughput when compared to the results of our previous research. Nevertheless, there is still scope for further enhancement. By analyzing the topology type and network congestion, we can identify potential bottlenecks that may limit

the throughput of the system. Furthermore, we can explore various optimization techniques to improve the overall performance of the system, such as load balancing and dynamic routing. By addressing these issues, we can further improve the throughput of the system and enhance its overall efficiency.

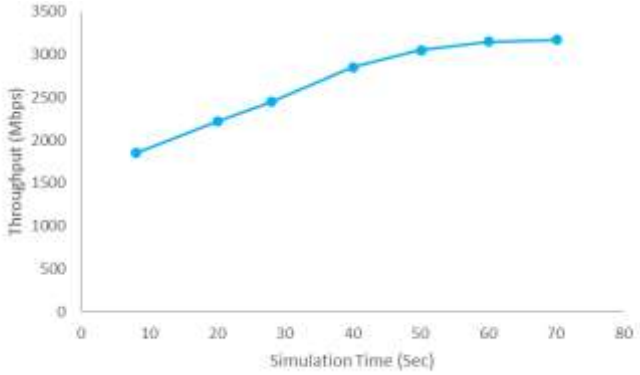


Fig. 12. Throughput of proposed framework at various simulation time

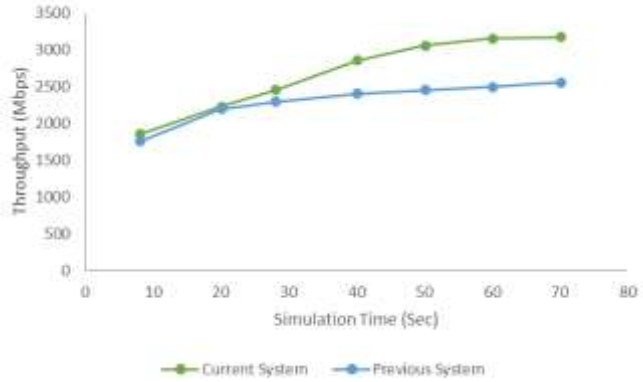


Fig. 13. Throughput comparison of Current system versus Previous system

4.4 Performance analysis

The results of the experiments conducted demonstrate that the finetune algorithm proposed in this study outperforms other well-known algorithms with respect to important metrics such as energy consumption, network throughput, and packet latency. The proposed protocol plays a crucial role in addressing challenges associated with energy management and security for the next generation of IoT systems.

Ad-hoc On-demand Distance Vector (AODV) is a reactive routing protocol, which means it establishes a route to a destination only when it is needed. AODV uses a distance vector approach to find routes and it

maintains routing tables at each node. However, it suffers from issues such as scalability and security vulnerabilities.

Ad-hoc On-demand Multipath Distance Vector (AOMDV) is an extension of the Ad-hoc On-demand Distance Vector (AODV) routing protocol that enables multiple loop-free alternate paths between a source and destination node in a mobile ad-hoc network (MANET). AOMDV is suitable for delay-sensitive applications such as voice and video, as it can provide reliable and timely data transmission in highly dynamic network scenarios.

Destination-Sequenced Distance Vector (DSDV) is a distance vector routing protocol, which means that each node only maintains information about its immediate neighbours and the routes to other nodes are determined based on the distance to these neighbours. It uses hop count as the metric for measuring the distance between nodes. However, it suffers from the problem of route loops due to the slow convergence of the protocol in large networks. Additionally, the frequent route updates can cause a significant amount of network overhead.

Energy efficient secured cluster based distributed fault diagnosis (EESCFD) uses an energy-efficient communication protocol to reduce energy consumption during the fault diagnosis process. EESCFD aims to achieve a balance between energy efficiency, fault diagnosis accuracy, and security in WSNs. The EESCFD approach is limited in scalability due to the size of the network. As the network grows in size, the number of clusters and nodes also increase, which can result in performance degradation.

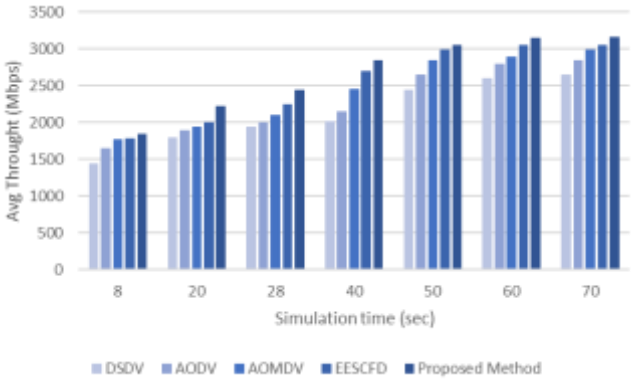


Fig. 14. Comparison of the throughput between the proposed method and other routing algorithms

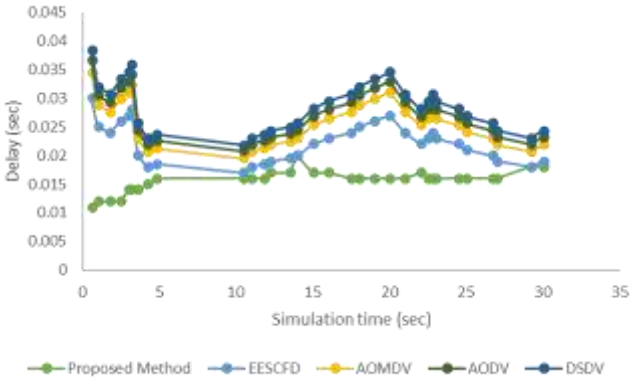


Fig. 15. Comparison of the delay between the proposed method and other routing algorithms

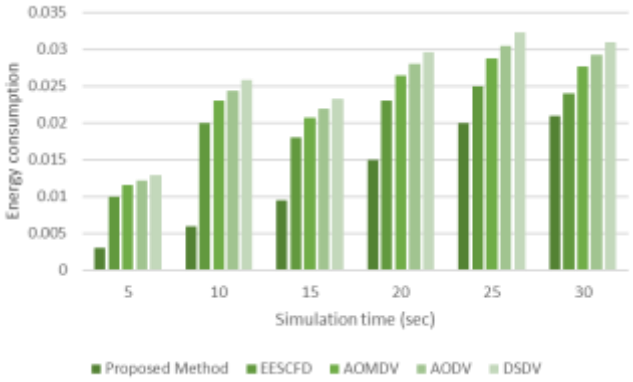


Fig. 16. Comparison of the energy consumption between the proposed method and other routing algorithms

5 Conclusıon & Future Scope

In conclusion, this research proposed a novel blockchain-based mechanism that enables secure and efficient communication among IoT devices in an SDN-enabled network with NFT integration. The proposed approach uses NFTs to represent the ownership and authenticity of devices, ensuring secure communication among them. The experimental results showed that the proposed mechanism outperforms the existing state-of-the-art methods in terms of energy consumption, throughput, and latency.

However, there is still room for improvement in the proposed mechanism. One potential area of future research is to investigate the impact of different topology types and network congestions on the performance of the proposed mechanism. Additionally, more in-depth analysis can be conducted to evaluate the proposed mechanism's security and privacy features under various attack scenarios. Finally, the proposed mechanism

can be further optimized for specific IoT applications, such as healthcare and smart home, by considering their unique requirements and constraints.

References

1. N. M. Karie, N. M. Sahri and P. Haskell-Dowland, "IoT Threat Detection Advances, Challenges and Future Directions," 2020 Workshop on Emerging Technologies for Security in IoT (ETSecIoT), Sydney, NSW, Australia, 2020, pp. 22-29, doi: 10.1109/ETSecIoT50046.2020.00009.
2. HP. (n.d.). Internet of things security primer. Retrieved from https://www.hp.com/us-en/shop/tech-takes/internet-of-things-security-primer.
3. Grove, R. (2021, June 29). Healthcare industry cyberattacks increase by 45%. HIPAA Journal. Retrieved from https://www.hipaajournal.com/healthcare-industry-cyberattacks-increase-by-45/
4. IBM. (2018). A blockchain platform for high-performance supply chain management. https://www.ibm.com/downloads/cas/OJDVQGRY
5. Unit 42. (2020). IoT Threat Report 2020. Palo Alto Networks. Retrieved from https://unit42.paloaltonetworks.com/iot-threat-report-2020/
6. Palo Alto Networks. (2021). The State of IoT Security in the Connected Enterprise: A Research Report. Retrieved from https://www.paloaltonetworks.com/apps/pan/public/downloadResource?pagePath=/content/pan/en_US/resources/research/connected-enterprise-iot-security-report-2021
7. Z. Zhang, R. Wang, X. Cai and Z. Jia, "An SDN-based Network Architecture for Internet of Things," 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Exeter, UK, 2018, pp. 980-985, doi: 10.1109/HPCC/SmartCity/DSS.2018.00162.
8. Li, J., Ma, X., Li, J., & Jiang, X. (2020). A blockchain-based approach to enhance security and privacy in industrial Internet of Things. IEEE Transactions on Industrial Informatics, 16(1), 69-78.
9. Zhang, Y., Liu, Y., Zhou, M., & Zhang, Y. (2019). A survey on blockchain-based secure communication for Internet of

Things. Journal of Network and Computer Applications, 125, 1-18.

10. Apte, P., & Arvind, T. (2021). A blockchain-based solution for IoT security and privacy using NFTs. International Journal of Intelligent Systems and Applications, 13(2), 15-23.

11. H. Zhou, J. Chen, & J. Lü. (2020). Secure and efficient data sharing in edge computing based on blockchain and NFT. Journal of Parallel and Distributed Computing, 146, 197-205.

12. Zang, Y., Cai, J., Zhu, X., & Hu, J. (2021). Secure data sharing of IoT enabled devices based on blockchain and smart contract. IEEE Access, 9, 74690-74701.

13. A. Al-Fuqaha, A. Khreishah, M. Guizani, A. Rayes and M. Mohammadi, "Toward better horizontal integration among IoT services," in IEEE Communications Magazine, vol. 53, no. 9, pp. 72-79, September 2015, doi: 10.1109/MCOM.2015.7263375.

14. C. Qiu, F. R. Yu, H. Yao, C. Jiang, F. Xu and C. Zhao, "Blockchain-Based Software-Defined Industrial Internet of Things: A Dueling Deep ${Q}$ -Learning Approach," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4627-4639, June 2019, doi: 10.1109/JIOT.2018.2871394.

15. Javier. (2021). Secure Combination of IoT and Blockchain by Physically Binding IoT Devices to Smart Non-Fungible Tokens Using PUFs. MDPI.

16. S. Bradić, D. Delija, G. Sirovatka and M. Žagar, "Creating own NFT token using erc721 standard and solidity programming language," 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 2022, pp. 1053-1056, doi: 10.23919/MIPRO55190.2022.9803593.

17. C. Karapapas, G. Syros, I. Pittaras and G. C. Polyzos, "Decentralized NFT-based Evolvable Games," 2022 4th Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), Paris, France, 2022, pp. 67-74, doi: 10.1109/BRAINS55737.2022.9909178.