Securing Patient Data in IoT Devices: A Blockchain-NFT Approach for Privacy, Security, and Authentication

Farha Masroor[a], Adarsh Gopalakrishnan[b] and Neena Goveas[c]
CSIS Department, BITS Pilani Goa Campus, Goa, India

Keywords: Healthcare Management, IoT, Blockchain, Security, Microcontrollers, Non-Fungible Tokens (NFT), Zero-Trust Type Architecture (ZTA).

Abstract: The Internet of Things (IoT) is poised to revolutionize healthcare by enabling remote patient monitoring and data access from any device. However, ensuring secure and flexible access control on IoT device-based systems remains a challenge, especially when handling multiple users with varying privileges at different points in time. Most of the current proprietary IoT products have a data pipeline where the measured data is sent to cloud-based servers before any access is possible. This creates data privacy issues, access problems or devices not working when Internet connectivity is not perfect. An ideal solution for an IoT-based system is one in which data can be stored and accessed on device in real-time with additional cloud-based storage and access for later use. To address these challenges, we propose a Blockchain-based Non-Fungible Token (NFT) based mechanism for IoT systems. Our system uses Blockchain technology to provide untameable NFT access keys, ensuring only authorized individuals can access patient data on a given IoT device. We conducted an experimental study using ESP32 microcontroller, Beaglebone Black boards, and Raspberry Pi devices to evaluate the effectiveness of our approach. Our results show that this approach is suitable for deployment on resource-constrained devices, with minimal computational requirements and negligible delays. Additionally, we implemented a Zero-trust type of architecture where no implicit trust is granted to any user or device, regardless of prior successful authentication and authorization validation. We find that the delays due to the additional processing of NFTs are negligible even within such constraints. Our findings demonstrate that utilizing NFTs for access control of patient data on resource-constrained IoT devices is feasible and offers a secure and scalable solution for developing cost-effective and safe IoT systems for healthcare.

1 INTRODUCTION

IoT-based devices are crucial for pervasive healthcare systems (Hwang and Lee, 2020), but they face challenges ensuring secure data sharing. Researchers (Islam et al., 2020), (Sharma et al., 2021) show that IoT devices embedded with sensors in wearable devices or within our environment can revolutionize healthcare. However, securing and sharing sensitive healthcare data remains a key concern. Blockchain is proposed as a solution to ensure secure access and management of medical records (Li and Dun, 2020). Blockchain, with its immutability and consensus mechanism (Cunha et al., 2021), has the potential to provide a trusted environment.

In this paper, we propose a Blockchain NFT-IoT system that addresses the above challenges and prioritizes safe access to patient data.

This paper is structured as follows: in Section 2, we provide an overview of existing work. Section 3 explores fundamental Blockchain concepts. Following this, in Section 4, we discuss our mechanism for securing healthcare data in IoT devices using Blockchain-NFT. Section 5 elaborates on an application example. In Section 6, details about the dataset used in our study are provided. In Section 7 we present the results. Lastly, Section 8 summarizes our conclusions and outlines future work.

2 RELATED WORK

Researchers have explored enhancing IoT security with Blockchain solutions. Kotz et al. in (Kotz and Peters, 2017) highlighted challenges in IoT technol-

[a] https://orcid.org/0009-0008-1314-4923
[b] https://orcid.org/0009-0007-0083-5891
[c] https://orcid.org/0000-0002-5395-4962

Masroor, F., Gopalakrishnan, A. and Goveas, N.
Securing Patient Data in IoT Devices: A Blockchain-NFT Approach for Privacy, Security, and Authentication.
DOI: 10.5220/0012764800003767

In *Proceedings of the 21st International Conference on Security and Cryptography (SECRYPT 2024)*, pages 704-709
ISBN: 978-989-758-709-2; ISSN: 2184-7711

ogy like secure ownership and managing access for multiple users. Dorri et al. proposed a Blockchain-based method for secure smart homes, but it may be impractical for widespread use (Dorri et al., 2017). An NFT-based approach was introduced for IoT access control by Sharma et al. (Sharma and Goveas, 2023), validated on ESP32 and Raspberry Pi devices. Ali et al. noted Blockchain's potential in decentralizing IoT data management (Ali et al., 2018).

Saeed et. al. discussed significant challenges in medical care data, such as integrity, manipulation, and fraud, highlighting Blockchain as a solution (Saeed et al., 2022). Abbas et al. reviewed 53 articles from 2016 to 2021 on Blockchain in healthcare, highlighting advantages like data integrity, participant anonymity, and improved drug supply chain management (Abbas et al., 2022). Atlam et al. (Atlam et al., 2020) demonstrated Blockchain's potential for secure identity management. Alamri et al. reviewed Blockchain for IoT, focusing on privacy and access control challenges in (Alamri et al., 2019).

3 CORE CONCEPTS OF BLOCKCHAIN

For this work, we have used the Solana Blockchain as it provides many features that are user-friendly and suitable for an IoT implementation

3.1 Solana Blockchain

Solana is a Blockchain platform that has been designed to provide a decentralized infrastructure for decentralized applications (dApps). It strives to offer a secure, fast, and scalable environment for dApps. Solana implements a unique combination of technologies including a Proof of History (PoH) consensus mechanism based on proof-of-stake (PoS) that helps to enhance scalability and efficiency.

For our prototype, Solana is great to build on due to its unique NFT Programs, low Transaction Costs, scalability, developer-Friendly Environment and Devnet which allows developers to test applications in a simulated environment, ensuring a secure and reliable system.

3.2 Solana Wallet

A Solana wallet is a digital wallet used for securely storing, sending, and receiving the native SOL tokens on the Solana Blockchain. It incorporates a private key, which is an exclusive sequence of characters utilized to authenticate transactions on the blockchain, guaranteeing their security and validity. Solana Wallets can be created and managed using various tools, such as browser extensions, mobile apps, and desktop software. Solflare and Phantom are some popular Solana wallets. Solana wallet is important for interacting with the Solana Blockchain and participating in decentralized finance (DeFi) applications and other blockchain-based services.

Token Metadata program: The Token Metadata Program (TMP) on the Solana blockchain is crucial for managing NFTs. It stores and manages metadata associated with NFTs, including their name, description, image or media files, external URLs, and other relevant attributes. This program facilitates the creation, storage, and retrieval of metadata, enhancing the usability and value of NFTs on the Solana blockchain.

Solana NFTs: Solana NFTs are created through Mint Accounts, each with unique attributes and configured with a supply of 1, ensuring only one token is issued and cannot be divided. These NFTs lack mint authority, making them non-fungible and preventing the creation of additional tokens. Solana's Token Metadata program includes Master Edition NFTs, the original NFTs, and Print Edition NFTs, used to manage and create copies of a specific NFT.

Master Edition NFTs are the original NFTs, created with specific attributes and containing metadata describing the original digital asset. They are unique and distinct, with the Mint Account associated with the Master Edition having the authority to mint new instances or editions based on it. The Master Edition account can also enable users to create multiple copies of the NFTs, with an attribute called "Max Supply" defining the upper limit of copies that can be created.

Print Edition NFTs are derived from Master Edition NFTs, serving as the primary copies of the original digital asset. They inherit some metadata and properties from the Master Edition but are distinct NFTs. The Master Edition NFT specifies the attributes of the printed editions, such as the number of copies that can be created and any restrictions on the copies. Printed Edition NFTs are directly linked to and derived from Master Edition NFTs, providing a way to create copies of the original asset.

4 SYSTEM MODEL OVERVIEW

4.1 Prototype of Our IoT System

Software developed for IoT systems needs to be used on and with resource-constrained devices. To

showcase our system's functionality, we've selected popular boards: ESP32 microcontroller, Beaglebone blackboard, and the Raspberry Pi 3.

4.1.1 Configuring a Web Server on ESP32 Microcontroller

To prepare our ESP32, we first configure the ESP32 board and install the ESP32 Espressif package into our Arduino IDE (url, 2024a). Next, we incorporate the source code for configuring the website server on the ESP32 microcontroller. The WiFi module establishes a connection to any WiFi hotspot, enabling the server to test. Upon setup completion, the server provides an IP address that serves as the API endpoint for our web platform.

4.1.2 Configuring a Web Server for Raspberry Pi and Beaglebone Black Boards

To set up a web server on a Raspberry Pi and Beaglebone black, we utilize the Python Flask framework (url, 2024b).To start, we install the Raspbian or Debian Linux-based OS, along with Python and the web framework named Flask, on both Beaglebone and the Raspberry Pi. Then, we develop a Flask-based web app and initiate it, by running the command "python3 app.py." To test the web server, we would access it from a browser on a different device, using the IP address of the Raspberry Pi or Beaglebone. Furthermore, we customize the web server as required by utilizing Flask's settings and options.

4.1.3 NFT and Edition Management for User Access on Our Web Platform

Our web platform is designed with a Next.js interface, allowing integration of users with their wallets through the @solana/web3.js package. In addition to granting access to all devices and storing the Master Edition NFTs, the advertisement wallet is also in charge of creating these NFTs. The software platform generates NFTs using an industry-standard json structure, and the metadata includes the device's MAC address. This MAC Address is added to the NFT's characteristics array for additional trust that the Non-Fungible tokens in the user's hands match the selected device. The platform uploads NFT to a decentralized database such as Arweave using Sugar CLI, an interface for the command line developed by the Metaplex Foundation, and mints NFT using the Metaplex JS SDK. Users may easily access and manage NFTs related to their devices by minting Edition NFTs from the admin wallet using the Metaplex JS SDK, which facilitates efficient NFT administration and distribution.

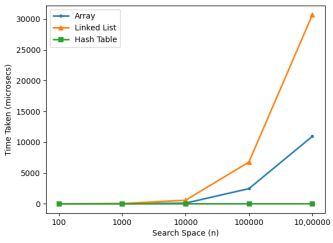
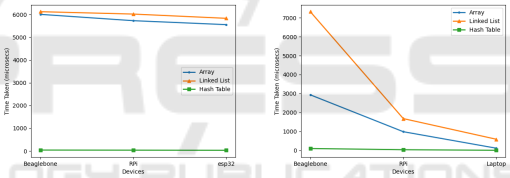


Figure 1: Time taken for NFT search Across Different Data Structures with Varying Search Space sizes (n).

4.2 Deployment of the Prototype

It is necessary for users to link their Solana Wallet to the website. Once the link has been made, users may choose which device to utilize. Once a device and operation are chosen, the web platform verifies the presence of the accurate NFT in the wallet of the user and the NFT Metadata for the MAC Address is checked, matching it with the MAC Address of the device. In our prototype, both the devices and patient data are resources and require permission to access. After these checks, the user can access the device and the patient data on it.



(a) Search space = 1000. (b) Search space = 10,000.

Figure 2: Combination of the Devices and the Data Structures.

The admin wallet has the authority to mint a printed edition NFT of the master edition for every user in the system. Next, a URL link is produced and forwarded to the selected non-admin user. The non-admin user sees an option of renting the NFT. The NFT is added to their wallet after approval. This is the initial stage of getting access to the system's resources.

5 HEALTHCARE DEVICE AND ACCESS SYSTEM DESIGN

Using the system model outlined in 4, the resource-constrained devices, web application, Solana Wallet integration, NFT management, and device access control mechanisms work together to enable users to connect, manage, and interact with IoT devices through the web platform. This includes aspects such as

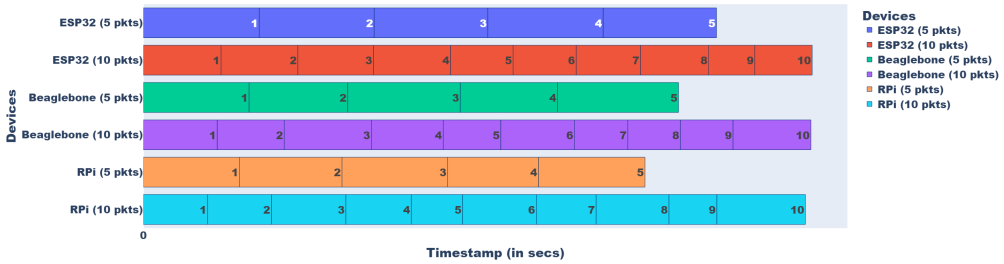


Figure 3: Pattern Analysis of Request-Response in Different Packet Sizes at Each Timestamp.

how devices are identified, NFTs minting and management, access control is implementation, and how users interact with the system through the web interface, including device connections, NFT creation, and deployment.

Patients are required to link their Solana Wallet to the web platform. Once connected, they can select the IoT device where they want to store their data. In our prototype deployment, we had three resource-constrained IoT devices. After selecting a device, the platform validates the presence of the correct NFT for that device in the patient's wallet. Upon successful validation, the patient can proceed to store their health data on the IoT Device. Each IoT device functions as a server, and each patient has their NFTs for identification, stored in their wallet. An NFT used and stored is of length 32 to 64 bytes as a public key.

Subsequently, medical professionals who need access to the patient's data can visit the web platform and connect their wallets to the application. The application retrieves their public key to determine which NFTs the medical professional owns. If the medical professional requires access to a specific device, the platform checks for the presence of the corresponding NFT. However, to access the patient's data, the medical professional must also possess the NFT specific to that patient, which is unique for each patient. The requirement of both NFTs for access adds an extra layer of security, as possession of only the device NFT is insufficient to access patient data.

If the medical professional has both NFTs, they can access the device and the patient's data stored on the device. If this authentication process needs to be completed only once, we call it a One-Time Verification architecture (OTV). Subsequently, the medical professional can access the data anytime and for any number of sessions without repeating the NFT verification process.

Additionally, another Zero-Trust type of Architecture has been implemented to enhance data security and reliability. Under this architecture, when a medical professional gains access to a device and a patient's data, the device is checked for authentication at the beginning of each session. Anytime, even during the session, if the medical professional requests patient data again, the NFT verification process executes. Any log out and log back in is treated as a fresh session and they must undergo the patient NFT verification process each time to access the data. This ensures that access is continuously authenticated and verified, enhancing the security of the system.

6 DATASET USED FOR TESTING OUR PROTOTYPE

We utilized ECG data from the MIT-BIH Arrhythmia database available on PhysioNet[1]. The dataset comprises 48 records, each containing CSV files and reference annotation files (Khincha et al., 2020).

For our system, we utilized 5 minutes of one patient record and stored it as a data input on the device.

In real life, sensors connected to the patient would collect the data, which would then be transferred to the devices for storage. Here we fed the selected sample data into the device for storage.

7 RESULTS

This section presents the results from our experiments and demonstrations NFT stored is of length 32 to 64 bytes, so storing and searching through them could be time-consuming on the devices selected for a larger number of records. We initiated our study by experimenting with different data structures and varying the search space size (n) that is the number of records 10, 100, 1000 up to 1000000 and obtained the time required for the search. The data structures included hash tables, arrays, and linked lists for storing the NFT token objects.

Figure 1 illustrates that the hash table exhibits significantly faster performance compared to arrays and linked lists. The search time remains relatively

[1]https://www.kaggle.com/datasets/mondejar/mitbih-database?resource=download

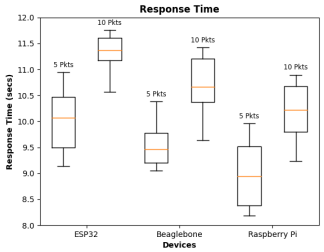
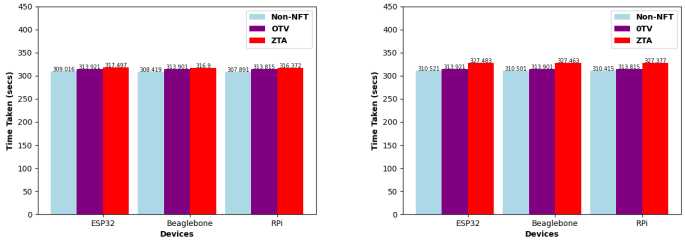


Figure 4: Response Time with varying Packet Sizes and Devices.



(a) With 5 packets. (b) With 10 packets.

Figure 5: Comparison plots of Architectures with varying Devices.

constant regardless of the search space (n), reflecting its O(1) access time. Conversely, arrays and linked lists show increasing search times with larger search spaces (n), with linked lists exhibiting the slowest performance due to their non-contiguous memory allocation.

We used a search space (n) of 1000 and 10,000. The experiment involved three resource-constrained devices (Raspberry Pi, Beaglebone, ESP32) and a laptop, combined with three different data structures (arrays, linked lists, hash tables). Each device and data structure handles objects differently, making it important to determine the most suitable data structure for storing objects like NFTs. Figure 2a illustrates the experiment with ESP32, Raspberry Pi, and Beaglebone, considering the memory constraints of ESP32, which cannot efficiently handle more than 1000 objects. In Figure 2b, we increased the search space to 10,000 and included a laptop along with Raspberry Pi and Beaglebone, using the same data structure combinations. This increase in search space allows us to observe noticeable delays and compare the behavior of the combinations with small and large search spaces. However, In our analysis depicted in Figure 2, we found that hash tables combined with a local machine exhibited the minimum delay and processed faster regardless of the search space (n) value. Additionally, we observed that using hash tables with any of the four devices for searching objects resulted in only slight differences. Conversely, linked lists and arrays were not optimal choices for storing and searching NFTs if the requirements of the prototype include time efficiency.

We studied the time delays of sending and receiving data between patient devices to the devices of medical professional. Initially, we segmented an ECG dataset, as illustrated in Section 6, and converted the CSV files to JSON format for storage on the devices. Subsequently, we created packets of two different sizes: 5 (each of 1 minute length) and 10 (each of 30 seconds length), by dividing the sample 5 minute ECG data.

The Gantt chart of the time delay of this process is shown in figure 3. The figure shows the timestamps indicating when each packet was sent and when it was received by the medical professional. This shows that variations in packet sizes and devices used.

We calculated the response time by averaging multiple requests and responses and plotted boxplots showing the minimum and maximum values for varying packet sizes with all three devices. Figure 4 demonstrates that Raspberry Pi has the shortest response time compared to other devices for both packet sizes

Once the comparison of packet sizes, devices, and data structures along with the search space was completed, we implemented our Blockchain-NFT-IoT system with One-Time Verification Architecture (OTV) and Zero-Trust type of Architecture (ZTA). We then compared both architectures with the non-NFT approach, where no security measures were implemented figure 5. We have used a hash table as the data structure for storing NFTs and have compared all three architectures with the three devices. The figure in Figure 5 indicates that Non-NFT has a shorter response time than OTV and ZTA for both the packet

size, as expected. However, the delays seen here are minimal due to the small search space (n) of 1000. If the search space (n) is increased to 10,000, delays will be larger for Rpi and Beaglebone. The delay difference between OTV and ZTA is not significantly high, making ZTA a feasible option.

8 CONCLUSION AND FUTURE WORK

In this work, we present an implementation of the Blockchain-NFT-IoT system on resource-constrained devices. We focused on evaluating and understanding the performance of different data structures, devices, and architectures in the context of a healthcare application involving the storage and transmission of ECG data. We found that even with resource-constrained devices like the ESP32, we can implement our proposed architecture. A Raspberry Pi can be used is there is a need for reduced response times for data transfer. Our implementation of One-Time Verification Architecture (OTV) and Zero-Trust type of Architecture (ZTA) demonstrated that the delays are not significant and they can be effectively used in resource-constrained devices thus ensuring data security in healthcare IoT environments.

This research is a first step in the design of secure IoT systems that are cost-effective and can provide real-time solutions even with poor connectivity. This can be of value when designing safe access mechanisms for healthcare and other sensitive data.

Future research directions could explore scaling these experiments to larger datasets experimenting with more devices and types of users and further enhancing security measures to meet the evolving needs of healthcare IoT applications.

REFERENCES

(2024a). Esp32 web server using arduino ide. https://randomnerdtutorials.com/esp32-http-get-post-arduino/. Accessed: (2024).

(2024b). Raspberry pi and beaglebone web server using flask. https://pythonbasics.org/flask-rest-api/. Accessed: (2024).

Abbas, A. F., Qureshi, N. A., Khan, N., Chandio, R., and Ali, J. (2022). The blockchain technologies in healthcare: Prospects, obstacles, and future recommendations; lessons learned from digitalization. *International Journal of Online & Biomedical Engineering*, 18(9).

Alamri, M., Jhanjhi, N., and Humayun, M. (2019). Blockchain for internet of things (iot) research issues challenges & future directions: A review. *Int. J. Comput. Sci. Netw. Secur*, 19(1):244–258.

Ali, M. S., Vecchio, M., Pincheira, M., Dolui, K., Antonelli, F., and Rehmani, M. H. (2018). Applications of blockchains in the internet of things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 21(2):1676–1717.

Atlam, H. F., Azad, M. A., Alzahrani, A. G., and Wills, G. (2020). A review of blockchain in internet of things and ai. *Big Data and Cognitive Computing*, 4(4):28.

Cunha, P. R. d., Soja, P., and Themistocleous, M. (2021). Blockchain for development: a guiding framework.

Dorri, A., Kanhere, S. S., Jurdak, R., and Gauravaram, P. (2017). Blockchain for iot security and privacy: The case study of a smart home. In *2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops)*, pages 618–623. IEEE.

Hwang, Y.-W. and Lee, I.-Y. (2020). A study on cp-abe-based medical data sharing system with key abuse prevention and verifiable outsourcing in the iomt environment. *Sensors*, 20(17):4934.

Islam, M. M., Rahaman, A., and Islam, M. R. (2020). Development of smart healthcare monitoring system in iot environment. *SN computer science*, 1:1–11.

Khincha, R., Krishnan, S., Parveen, R., and Goveas, N. (2020). Ecg signal analysis on an embedded device for sleep apnea detection. In *International Conference on Image and Signal Processing*, pages 377–384. Springer.

Kotz, D. and Peters, T. (2017). Challenges to ensuring human safety throughout the life-cycle of smart environments. In *Proceedings of the 1st ACM Workshop on the Internet of Safe Things*, pages 1–7.

Li, J. and Dun, W. (2020). Range query in blockchain-based data sharing model for electronic medical records. In *Journal of Physics: Conference Series*, volume 1634, page 012035. IOP Publishing.

Saeed, H., Malik, H., Bashir, U., Ahmad, A., Riaz, S., Ilyas, M., Bukhari, W. A., and Khan, M. I. A. (2022). Blockchain technology in healthcare: A systematic review. *Plos one*, 17(4):e0266462.

Sharma, N., Mangla, M., Mohanty, S. N., Gupta, D., Tiwari, P., Shorfuzzaman, M., and Rawashdeh, M. (2021). A smart ontology-based iot framework for remote patient monitoring. *Biomedical Signal Processing and Control*, 68:102717.

Sharma, R. K. and Goveas, N. (2023). Use of blockchain in securing iot systems with resource constrained devices. In *2023 IEEE 20th International Conference on Software Architecture Companion (ICSA-C)*, pages 216–223. IEEE.