








# Blockchain-Driven Hybrid Model for IoT Authentication

Truong Duy Dinh<sup>1</sup> , Tran Duc Le<sup>2</sup> , Khanh Quoc Dang<sup>2</sup> ,  
Vladimir Vishnevsky<sup>3</sup> , and Ruslan Kirichek<sup>4</sup> 

<sup>1</sup> Posts and Telecommunications Institute of Technology, Hanoi, Vietnam  
duydt@ptit.com.vn

<sup>2</sup> University of Science and Technology – The University of Danang, Danang,  
Vietnam  
letranduc@dut.udn.vn

<sup>3</sup> V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences,  
Moscow, Russia

<sup>4</sup> The Bonch-Bruевич  
Saint Petersburg State University of Telecommunications, Saint Petersburg, Russia  
kirichek@sut.ru

**Abstract.** The Internet of Things has the potential to play a significant part in the ongoing Industrial Revolution. Machines, devices, and sensors are able to connect and communicate with each other via networks. These Internet of Things devices generate an enormous amount of data that is sensitive to privacy and security. For this reason, the security of these devices is of the utmost importance to guarantee the system's reliability and efficiency. It has been suggested that one efficient way to increase the safety of authentication for Internet of Things networks is to use blockchain technology for the purpose of authenticating users and devices on those networks. The purpose of this study is to conduct research and make a proposal for a blockchain-driven hybrid model for the Internet of Things authentication. This model is intended to be an improvement based on previously developed models and has shown promising results on the Ganache blockchain.

**Keywords:** Internet of Things · authentication model · Blockchain

## 1 Introduction

In recent years, the Internet of Things (IoT) has received much attention, and it is expanding quickly due to the spread of communication technology and the introduction of enough gadgets [1]. Based on the Ericsson Mobility Report<sup>1</sup>, 550 million 5G subscriptions will be available in 2022, with 10% of all subscriptions located in Asia-Pacific. By 2022, over 29 billion connected devices are predicted, with approximately 18

---

<sup>1</sup> <https://www.ericsson.com/en/about-us/company-facts/ericsson-worldwide/india/authored-articles/ushering-in-A-better-connected-future>.

billion involved with IoT. According to the Internet of Things Team - IDC<sup>2</sup>, there will be 41,6 billion connected IoT devices by 2025 in the world. These examples highlight the importance of IoT devices in today's digital society [2]. Additionally, IoT applications such as healthcare, smart home, and agricultural applications show that IoT has appeared everywhere and in all aspects of life.

Besides that, growth potential, IoT faces many challenges, such as interoperability, compatibility, limited bandwidth, data complexity, data volume, and especially security issues. The leading IoT security concerns are authentication, authorization, integrity, availability, and privacy [3–6]. In this research, we focus on the authentication issue in IoT networks because the authentication of users and devices in this network is still dependent on a third party.

Authentication determines whether a user or a device that wants to access the system is a valid user or device. Authentication is an essential requirement because it determines the security of the system. It allows valid users to operate and prevents system access and resource usage on the system from unauthorized users [7–9].

Recently, blockchain has emerged as a technology that can solve the authentication problem transparently and securely. In 2008, Satoshi Nakamoto proposed the blockchain [10]. All committed transactions are maintained in a sequence of blocks on a blockchain, which might be viewed as a public ledger. This chain expands continually as additional blocks are added. The essential properties of blockchain technology are decentralization, persistence, anonymity, and audibility. Integrating numerous essential technologies, including cryptographic hash, digital signature (based on asymmetric cryptography), and distributed consensus mechanism, enables blockchain to operate in a decentralized context. By using blockchain technology, a transaction may be conducted decentralized. Consequently, blockchain may significantly reduce costs and increase efficiency.

There are three main types of blockchain structures:

- **Public:** Public blockchains are decentralized and open to all participants. Public blockchains provide all blockchain nodes equal rights to access the blockchain, produce new data blocks, and validate data blocks. Cryptocurrencies such as Bitcoin, Ethereum, and Litecoin are exchanged and mined on public blockchains.
- **Private:** Private blockchains, also known as managed blockchains, are permissioned blockchains administered by a single entity. A central authority determines who can be a node in a private blockchain. In addition, the central authority does not always provide each node equal permissions to carry out functions. Private blockchains are only partially decentralized due to their restricted public accessibility. Ripple, a business-to-business virtual currency exchange network, and Hyperledger are private blockchains.
- **Consortium:** Consortium blockchains are permissioned blockchains administered by a group of organizations, as opposed to a single organization in the case of private blockchains. Therefore, consortium blockchains are more decentralized than private blockchains, resulting in greater security.

---

<sup>2</sup> <https://www.businesswire.com/news/home/20190618005012/en/The-Growth-in-Connected-IoT-Devices-is-Expected-to-Generate-79.4ZB-of-Data-in-2025-According-to-A-New-IDC-Forecast>.

To solve the authentication problem using blockchain, we also need Smart Contract [11]. Smart contracts are blockchain-based programs that execute when specific criteria are satisfied. Typically, they are used to automate the execution of an agreement so that all parties may know the outcome instantly, without the need for an intermediary or any time lost. In addition, they can automate a process by initiating the subsequent operation when certain circumstances are satisfied.

First, developers create snippet codes of Smart Contract and compile them into bytecodes. The bytecodes are deployed to the blockchain for future calls. Each time a user performs a transaction with Smart Contract, the EVM (Ethereum Virtual Machine) virtual machines will execute the Smart Contract command lines corresponding to the calls and update new states to the blockchain.

In the IoT authentication model, processing speed is also an essential factor. Recent studies indicate that fog computing can be a promising solution to this problem [12]. Fog computing is a distributed architecture that deploys storage and processing components to the cloud's edge. It is used to expand cloud computing. As a decentralized computing infrastructure, fog computing helps to bridge the gap between the cloud and where data is generated and operated. The objective is to serve new applications with lower latency needs while processing data more effectively to reduce network expenses. By bringing fog computing closer to IoT devices, instead of using cloud computing to perform real-time analytics and leverage computing power, the user experience can be significantly improved.

Based on the required authentication elements for IoT networks, we can divide authentication methods into user authentication, device authentication, and user and device authentication. Where authentication aims to answer the questions: Is it a valid user or not? Does the user own the data which he or she accesses? How to distinguish a malicious user from a legitimate user? Has the added device been used somewhere else by another user? Does one device have access to the other device's data? In recent studies, many authentication models for IoT networks based on blockchain have been proposed. However, most models only solve one of two problems: user authentication or device authentication. In our study, the authentication model will be used for both the authentication of users and devices in the network. The proposed model uses a Ganache<sup>3</sup> blockchain network with Smart Contract, fog computing simulated by NodeJS server, cloud, IoT devices and a web server. IoT devices are simulated with their MAC addresses and exchange information when requested. The simulation results show the usability of the proposed authentication model. This model can overcome some disadvantages of non-blockchain-based authentication and solve a few other problems in previous models.

The remainder of this study is organized as follows: Sect. 2 analyses the existing authentication methods based on blockchain for IoT networks. In Sect. 3, we propose an authentication model for both IoT devices and users. Section 4 shows the results of simulation and testing. Finally, Sect. 5 will provide the conclusions of this study.

---

<sup>3</sup> <https://trufflesuite.com/ganache/>.

## 2 Related Works

The essential qualities of blockchain are its distributed design, immutability, indestructibility, and fault tolerance. Due to these properties, blockchain-based authentication provides a novel and suitable method for authenticating IoT networks. Traditionally, IoT network authentication was conducted by a third-party intermediary who retained all information and fully controlled the user's authentication.

The existing IoT networks need authentication of both users and devices. User authentication aims to determine whether a valid user wants to access the resources within the scope of the authorized access to prevent the user from tampering with access to sensitive information. The primary purpose of device authentication is to determine which devices are genuine and which are malicious. In order to get information that compromises the security of user data, malicious devices may imitate regular devices by sending access requests to the whole network. Due to the limited processing power and storage capacity of IoT devices, a proper authentication mechanism may aid in securing devices from unauthorized access by attackers or malicious devices.

There has been some investigation into leveraging blockchain technology to provide transparent identity identification for IoT devices and users. The following are a few notable articles.

D. Li, W. Peng, W. Deng and F. Gai in [13] suggested using blockchain to construct a secure, tamper-proof ledger for IoT devices. Each device's unique ID was kept on the blockchain, enabling devices to validate each other without a central authority. New devices must register on the blockchain network to connect. Once authorized, the device may join the IoT network and share P2P data with other devices. It allows device-to-device communication and authentication. This might harm the IoT network and P2P devices if they cannot self-protect. This architecture authenticates network devices but not users utilizing devices.

An authentication model that utilizes fog nodes, which decrease network device processing power, was introduced in the paper [14]. As the number of devices increases, device authentication becomes more complicated, making network expansion harder. In addition, the smart contract offers mappings between devices and fog nodes and a list of user authentications. A device is only connected with a fog node under this approach, but a fog node may be connected to several devices. When a user needs to access a particular IoT device, it must send a request to the smart contract requesting authorization for this purpose. The user must make an access request to the fog node after gaining authorization to access the target device.

The study [15] presented a blockchain-based device and user authentication mechanism using gateways. In this study, a user requests a blockchain smart contract address from the gateway to access user and privacy policy information. If the request is accepted, the user will send another request to the gateway, which records user and device data on the blockchain. Besides user authentication, IoT devices must be authenticated by an administrator or another device manager. This model includes the IoT devices, users, and blockchain-connected gateway. Devices share information or resources with other users, devices under a device policy (smart contract). The blockchain-connected gateway will examine the user's eligibility (permissions and authentication) to access the device's information and resources. Once accepted, the user controls the device. The

blockchain and deliver system data. The author assesses and implements the model on Ethereum's blockchain and analyzes its states and expenses.

R. Kabir, A. T. Hasan, M. R. Islam, and Y. Watanobe in [16] presented a system that allocated a unique identifier to each device. Data transaction records for each device using smart contracts were stored in a blockchain-based on the open-source project BigChainDB. This system consists of IoT devices, cloud servers and the blockchain network. IoT devices may be linked to one another, and smart contracts accompany them. The cloud stores and encrypts the data from these devices. Blockchain smart contracts will manage each access to a device by other IoT devices or users. In addition, this approach applies the Practical Byzantine Fault Tolerance (pBFT) consensus mechanism, which recognizes the aberrant synchronization of data in the blockchain ledger to improve the security and reliability of the system. However, the paper only proposed an authentication method for IoT devices and did not give the system's performance results.

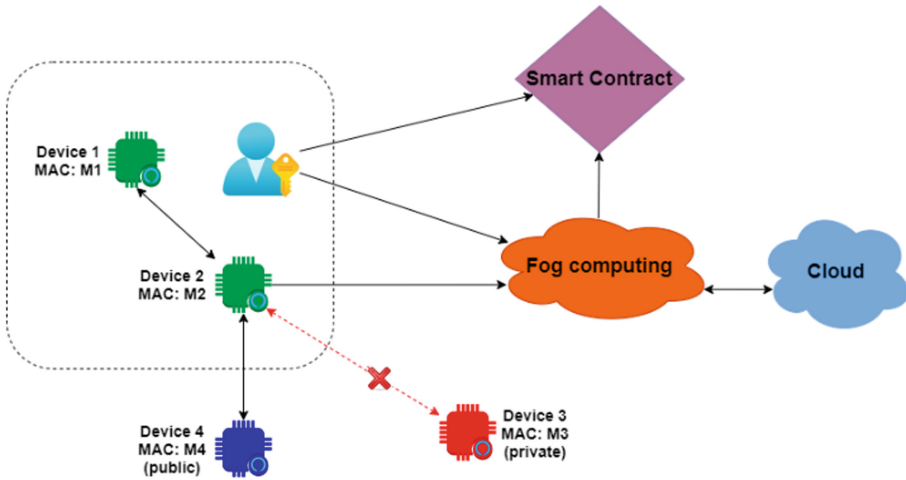
The study in [17] suggested a blockchain-based authentication strategy for the IoT system, which comprises Device Manufacturer, IoT devices, data centre servers, and blockchain. In particular, the modular square root (MSR) approach is utilized to ensure the security and efficiency of the authentication process, and blockchain technology is employed to increase security and offer scalability for this system. By using a smart contract, the system can ensure that only the registered devices or users can get the service to which they have subscribed. The model has four phases: system initialization, registration, authentication, and update and revocation. In order to evaluate the feasibility of the system, the authors provided an implementation of an Ethereum test network, Remix. However, the paper does not consider the authentication mechanism for users, which substantially impacts the system's security.

Gong-Guo Z. and Wan Z. [18] presented an IoT-chain security authentication solution using the Hyperledger Fabric blockchain platform. Inside the system, there are three sorts of chain codes: access code, device code, and policy code. The access code is the primary software that implements the user safety authentication procedure. The device code provides a query technique for the URL of the resource data offered by the storage device, while the policy code indicates the administrator user's access control strategy. Experimentation demonstrates that IoT-chain can sustain high throughput and achieve consensus efficiently in a distributed system. The study proposed authentication mechanisms for both IoT devices and users. Using Hyperledger Fabric, a private blockchain makes this system difficult to ensure the same level of security and transparency as a public blockchain. In addition, the data access rate is not mentioned, and IoT-chain operation diagram is not apparent.

### 3 The Proposed Authentication Model for IoT Devices and Users

The proposed model uses a Ganache blockchain network with Smart Contract, fog computing simulated by *NodeJS* server, cloud, IoT devices and a web server. IoT devices are simulated with their MAC addresses and exchange information when requested. Figure 1 presents the proposed model.

In the proposed model, we assume they are local devices owned by the same user (green colour: device 1, device 2). They can easily exchange information and get each



**Fig. 1.** The proposed authentication model for IoT network. (Color figure online)

other's data. There are other users' devices (red colour: device 3). These devices are private, and they do not share information. Another type of device is a public device (blue colour: device 4). Any user can access and get information from them.

### 3.1 Role of Smart Contract

A smart contract acts as an authentication centre storing information of users, devices, and relationships between users and devices. In this paper, Smart Contract is built using Solidity. The smart contract is compiled and migrated through Truffle. The compilation process will convert the Solidity code into bytecode and the Application Binary Interface (ABI). The bytecode will be used to deploy to the Ethereum network, while the ABI determines which function in the Smart Contract is called and returns the data in the expected format.

Migration is the process of converting compiled data onto the blockchain network. For Truffle, this process brings the compiled bytecode to a private blockchain network – specifically Ganache. Corresponding to each Smart Contract, an address will be used to identify it on the blockchain. This address is a long string of numbers and characters starting with 0x. Each address is associated with four different fields:

- Nonce: is an integer that is incremented every time the address sends any transaction.
- Balance: is the balance of the Smart Contract. If the user makes a transaction to this address, the balance will increase, and when the transaction is sent to another address, this balance will decrease.
- Code: source code of Smart Contract.
- Data: is where all the storage variables of a smart contract are stored. Memory variables cannot be stored in the blockchain.

### 3.2 Role of Fog Computing

We proposed to use fog computing as a bridge between users and devices. Fog computing will support storing data of devices and perform particular (pre-configured) calculation functions to reduce the computing pressure of the devices. Data such as MAC address is encrypted before being saved to the cloud. Every time a device or user wants to access another device, a request is sent to the Smart Contract to retrieve that device's information, and then compare it with the information submitted by the user/device to confirm the validation.

### 3.3 Simulation Model

The operating principle of the proposed model in Fig. 1 is shown through the simulation model. To build this model, we have some assumptions as follows:

- Cloud and fog computing will be simulated on the same server, which can receive and process user and device requests.
- The devices are simulated through the website service platform. They can receive and request data. The default settings are stored and exchanged with the server.
- The configuration of basic parameters on the device is saved as input values.

The operating principle of the proposed model is shown in Fig. 2.

We use *ReactJs* technology to support the user and device functions. There are some essential functions as follows:

- *getUsername, addInformation*: obtain and edit personal information including username and password;
- *addDevice, removeDevice*: add and remove devices;
- *getDeviceList*: get the entire list of devices owned by the current user;
- *switchDeviceStatus*: change the state of devices from private to public and vice versa;
- *getDeviceData*: retrieve data collected from other devices;
- *saveDataToCloud*: save collected information to the cloud.

The authentication model is based on asymmetric encryption using a public-private key pair generated and used as the user’s wallet address. Users can log in via Metamask wallet in the simulation model. Users need to store the mnemonic phrase for future account recovery. The password is also used to authenticate the user who owns this wallet properly. For web browsers that have never used this wallet address, users need to enter the mnemonic phrase to log in to the wallet. Figures 3 and 4 depict registering

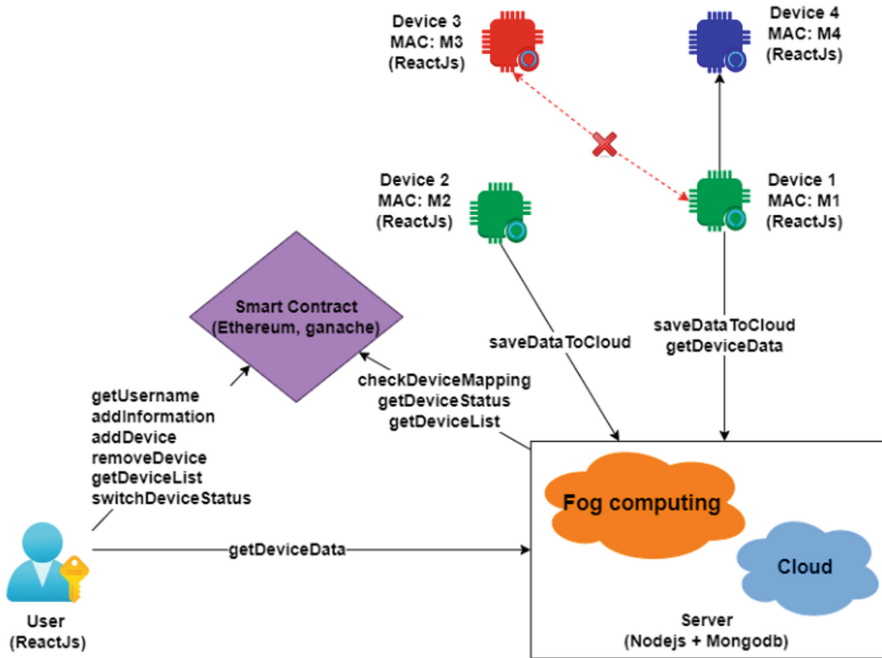


Fig. 2. Simulation model

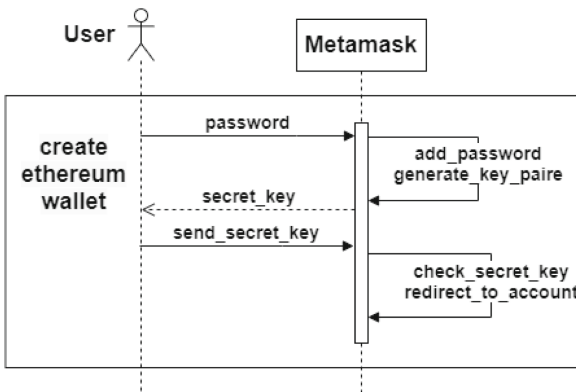


Fig. 3. Register a user account on the metamask e-wallet.



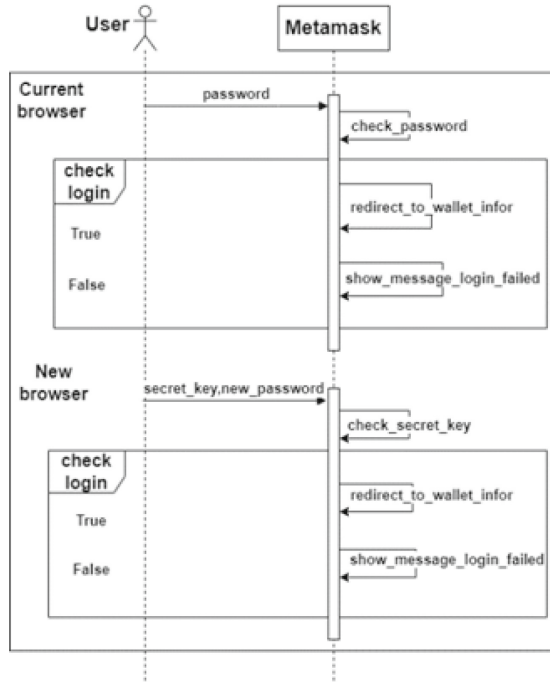


Fig. 4. Login to a metamask wallet on different web browsers.

a Metamask wallet and logging into it on one or more different browsers. There are some main functions: adding a new device, storing data, a user accesses the data of the device, a device gets data from other devices. In Figs. 5, 6, 7 and 8 below, the requests and responses in red are on-chain execution, and the requests and responses in black are off-chain execution.

**Adding a New Device:** Smart Contract requires the user to enter the name and MAC address of the device and send it up as a hash. The smart contract will check the uniqueness of this MAC address and add the device to the list with the user’s wallet address if this is a new device. The sequence diagram of device authentication when a user adds a new device is shown in Fig. 5.

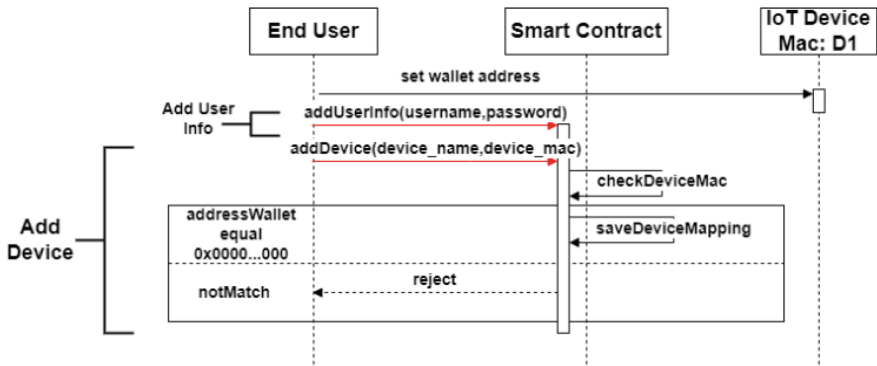


Fig. 5. Sequence diagram of device authentication when adding a new device.

**A Device Stores Data:** The device sends the collected data to fog computing along with its wallet address and MAC address. Fog computing will hash the device’s MAC address and get the corresponding wallet address stored on the blockchain. Fog computing will check the match between the wallet address of the device owner and the wallet address stored on the blockchain. Fog computing saves the device’s data to the cloud if they match. The sequence diagram for device authentication when a device stores data is shown in Fig. 6.

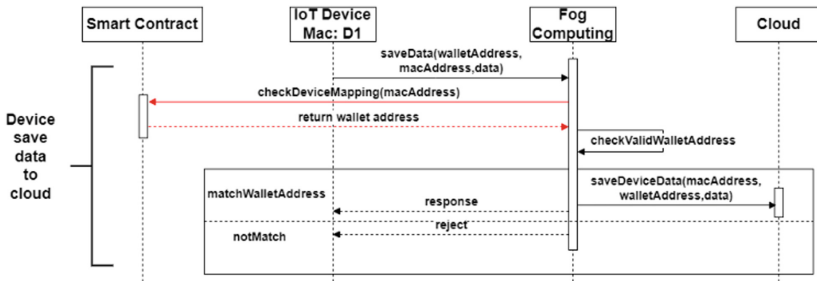


Fig. 6. Sequence diagram for device authentication when storing data.

**User Accesses the Device’s Data:** The user must simultaneously send the device’s wallet address and MAC address to fog computing to retrieve data from that device. Fog computing takes the list of MAC addresses corresponding to the wallet address and

matches it. If the MAC address exists in the list, fog computing will take the data in the cloud and return it to the user. The sequence diagram of user and device authentication when a user accesses device data is shown in Fig. 7.

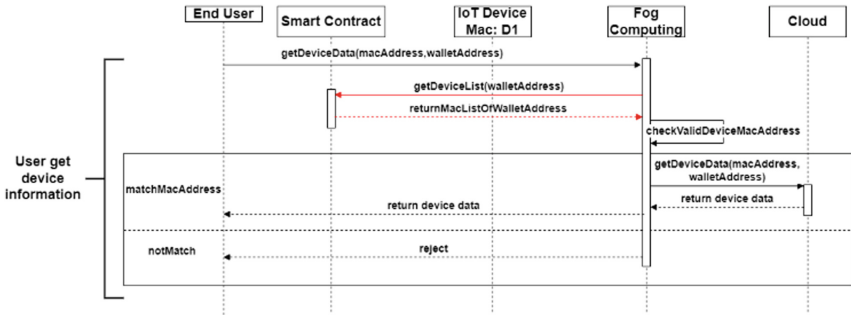


Fig. 7. Sequence diagram of user and device authentication when a user accesses device data.

**A Device Gets Data from Other Devices:** In this case, fog computing will check whether the data-retrieved device is a public device or has the same wallet address as the current device. If valid, it will return data from the cloud. If the data-retrieved device is private or another user owns it, the data cannot be obtained. The sequence diagram of device authentication when a device accesses data from another device is shown in Fig. 8.

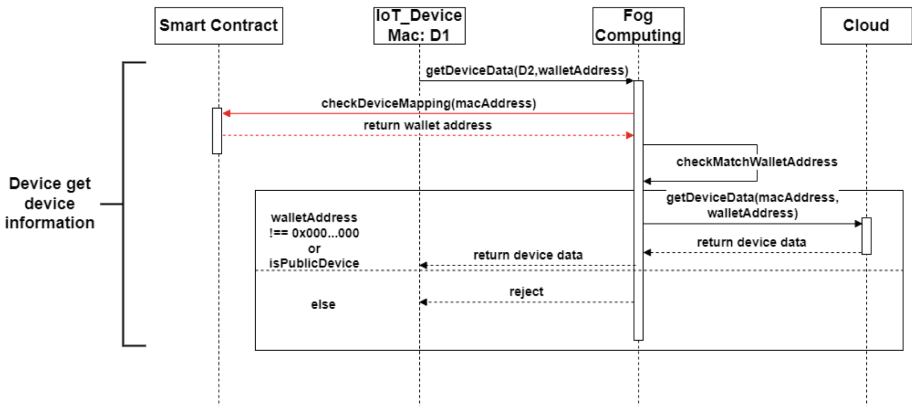


Fig. 8. Sequence diagram of device authentication when a device accesses data from another device.

## 4 Evaluation and Results

In this section, the testing result of the proposed model is presented. The Smart Contract is deployed with *Truffle* in Fig. 9. **CREATION TX** is the ID of a transaction. **STORAGE**

is the memory of Smart Contract that stores value variables. **TRANSACTION** is the details of transactions performed with Smart Contract.

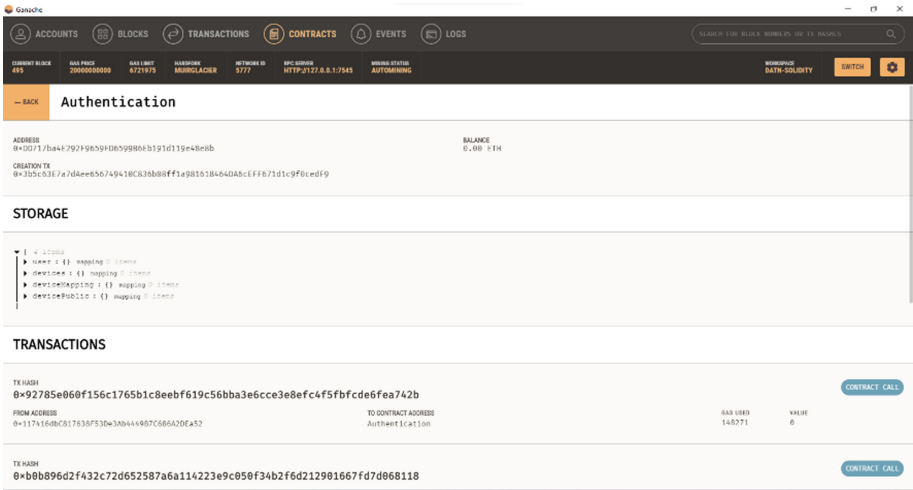


Fig. 9. Smart Contract on Ganache after deploying using Truffle.

In this scenario, we add a new device with a MAC address is 54:52:00:61:34:77 (Fig. 10). This transaction is confirmed in Fig. 11 and Fig. 12.

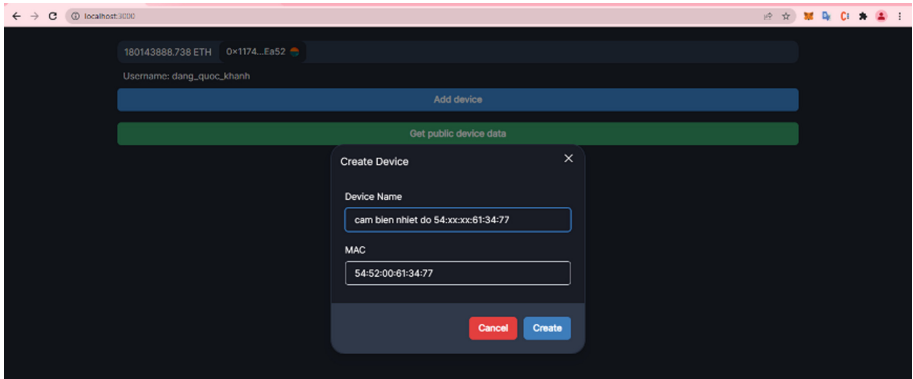


Fig. 10. Adding a new device.

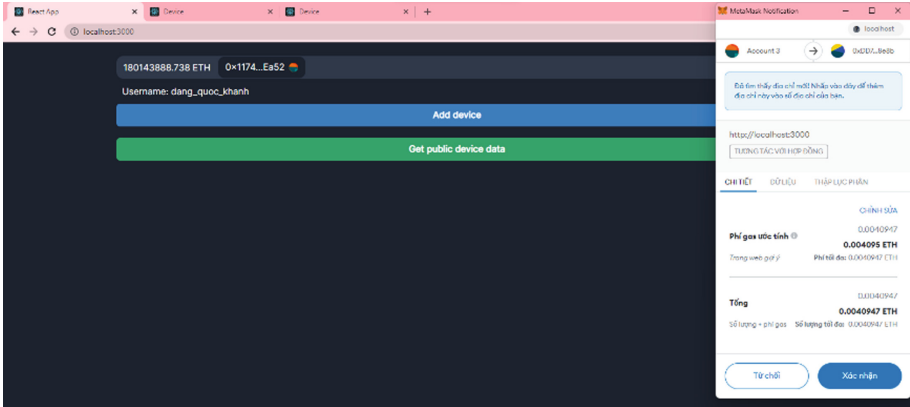


Fig. 11. Confirmation for the transaction of adding a new device.

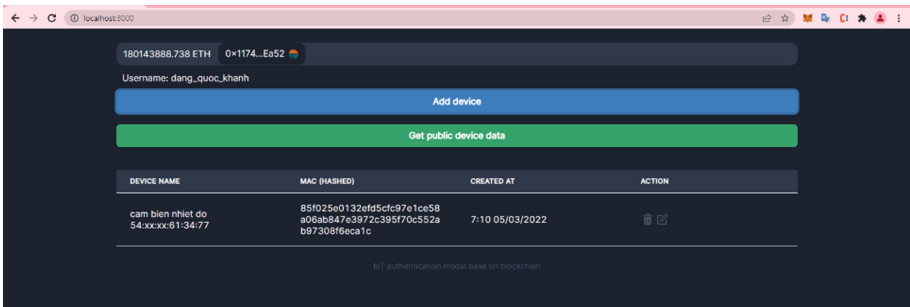


Fig. 12. Successfully adding device.

For simulation, we set the owner's wallet address (user `dang_quoc_khanh`) and manual data for the device to store in the cloud (Fig. 13).

Figure 14 shows the scenario when the user retrieves the stored data.

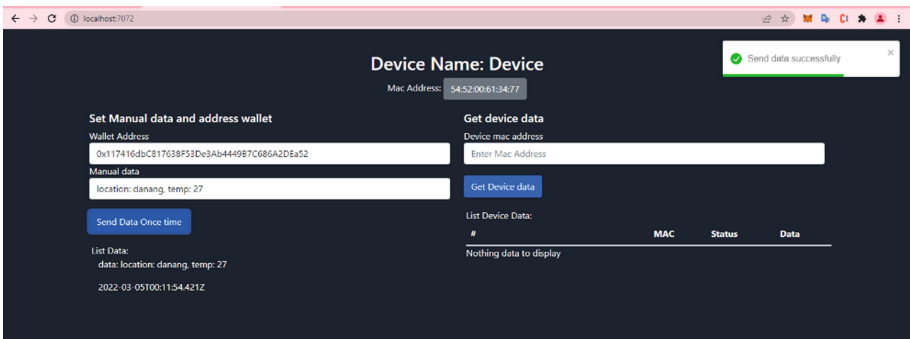


Fig. 13. The device enters the owner's wallet address and stores the data.

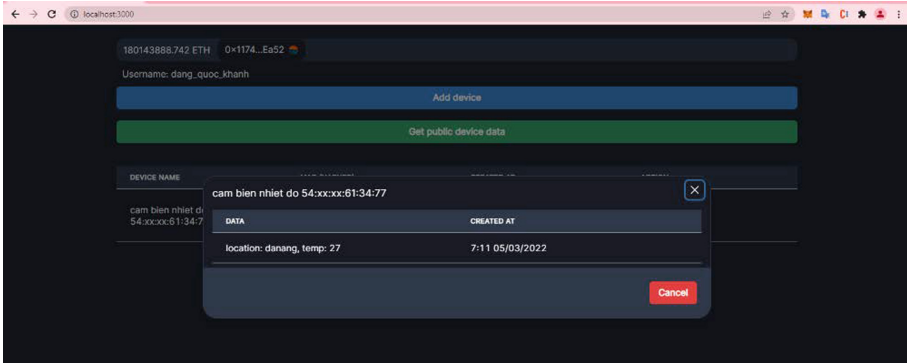


Fig. 14. The user retrieves the stored data.

We can set a device (MAC: 54:52:00:61:34:77) to public (Fig. 15). Then another user or device can access the data on this kind of device. For example, in Fig. 16, the device with MAC: 54:52:00:e3:75:79 can obtain data from this public device.

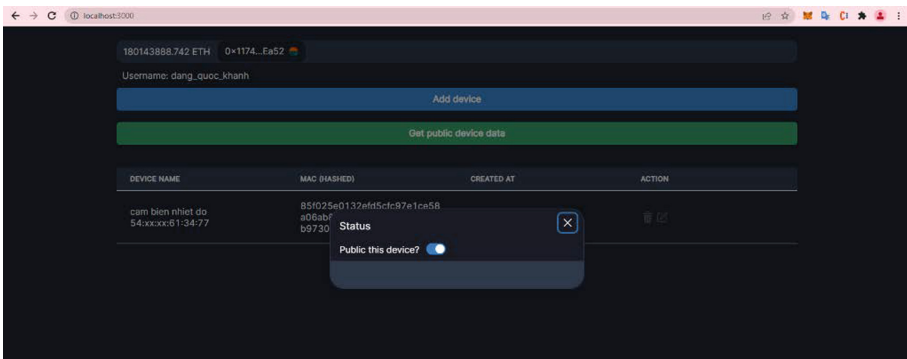


Fig. 15. Setting a device to public.

To demonstrate the function of preventing getting data from a private device, we set a new user named user\_guest. This user can get the data of a public device (MAC: 54:52:00:61:34:77) (Fig. 16). However, user\_guest cannot obtain the data of private data (MAC: 54:52:00:b1:c9:97) of the user dang\_quoc\_khanh (Fig. 17). Other devices (MAC: 54:52:00:e3:75:79) also cannot access private device data (Fig. 18).

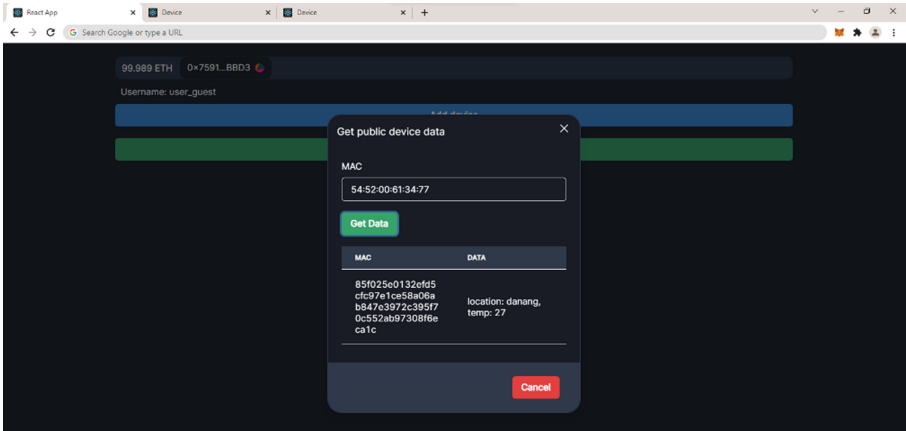


Fig. 16. Obtaining data from a public device.

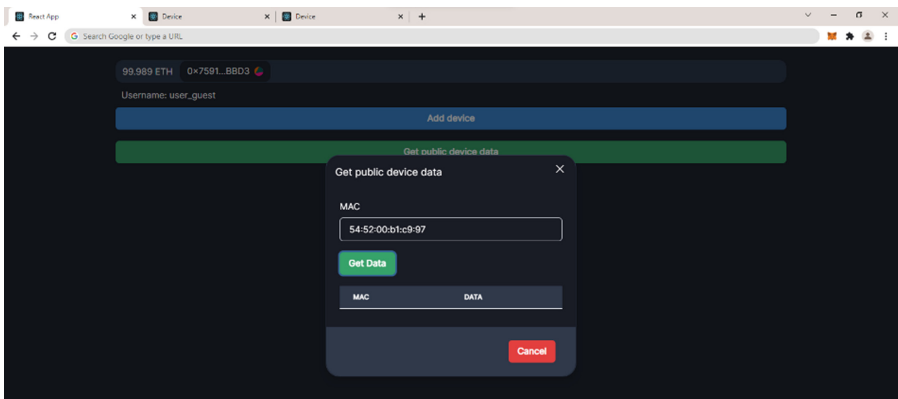


Fig. 17. No data is returned if the device is private.

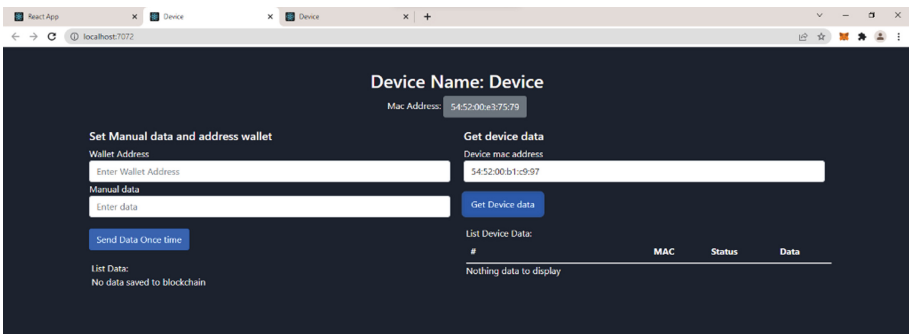


Fig. 18. Other devices also cannot access the data of a private device.

## 5 Conclusions

In this study, we proposed a blockchain-driven hybrid model for IoT authentication that is improved based on existing models. The model supports both user authentication and device authentication. It is simulated and tested by implementing a small system that applies a Ganache blockchain network with Smart Contract, fog computing simulated by NodeJS server, cloud, IoT devices and a web server. IoT devices are simulated with their MAC addresses and exchange information when requested.

We presented the authentication stages of IoT devices and users with sequence diagrams. Moreover, the validation process of the proposed model for each role and function was simulated. In addition, the transfer of a device owned by one user to another is taken into account and done easily. This leads to the conclusion that the system with device and user authentication outperforms a single user or device authentication.

**Acknowledgements.** The study was financially supported by the Russian Science Foundation within of scientific project No. 22-49-02023 “Development and study of methods for obtaining the reliability of tethered high-altitude unmanned telecommunication platforms of a new generation”.

## References

1. Reyna, A., Martín, C., Chen, J., Soler, E., Díaz, M.: On blockchain and its integration with IoT. Challenges and opportunities. *Future Gener. Comput. Syst.* **88**, 173–190 (2018)
2. Janarthanan, T., Bagheri, M., Zargari, S.: IoT Forensics: an overview of the current issues and challenges. In: Montasari, R., Jahankhani, H., Hill, R., Parkinson, S. (eds.) *Digital Forensic Investigation of Internet of Things (IoT) Devices*. ASTSA, pp. 223–254. Springer, Cham (2021). [https://doi.org/10.1007/978-3-030-60425-7\\_10](https://doi.org/10.1007/978-3-030-60425-7_10)
3. El-Hajj, M., Fadlallah, A., Chamoun, M., Serhrouchni, A.: A survey of internet of things (IoT) authentication schemes. *Sensors* **19**(5), 1141, 1–43 (2019)
4. Sazonov, D., Kirichek, R.: Digital object architecture as an approach to identifying Internet of Things devices. In: Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V. (eds.) *DCCN 2019. CCIS*, vol. 1141, pp. 597–611. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-36625-4\\_48](https://doi.org/10.1007/978-3-030-36625-4_48)
5. Al-Bahri, M., Ruslan, K., Aleksey, B.: Integrating internet of things with the digital object architecture. In: Galinina, O., Andreev, S., Balandin, S., Koucheryavy, Y. (eds.) *NEW2AN ruSMART 2019. LNCS*, vol. 11660, pp. 540–547. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-30859-9\\_47](https://doi.org/10.1007/978-3-030-30859-9_47)
6. Pomogalova, A., Sazonov, D., Donskov, E., Borodin, A., Kirichek, R.: Identification method for endpoint devices on low-power wide-area networks using digital object architecture with blockchain technology integration. In: Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V. (eds.) *DCCN 2021. LNCS*, vol. 13144, pp. 103–114. Springer, Cham (2021). [https://doi.org/10.1007/978-3-030-92507-9\\_10](https://doi.org/10.1007/978-3-030-92507-9_10)
7. Vladimirov, S., Kirichek, R.: The IoT identification procedure based on the degraded flash memory sector. In: Galinina, O., Andreev, S., Balandin, S., Koucheryavy, Y. (eds.) *ruSMART NsCC NEW2AN 2017. LNCS*, vol. 10531, pp. 66–74. Springer, Cham (2017)
8. Vladimirov, S.S., Pirmagomedov, R., Kirichek, R., Koucheryavy, A.: Unique degradation of flash memory as an identifier of ICT device. *IEEE Access* **7**, 107626–107634 (2019)



9. Al-Bahri, M., Yankovsky, A., Borodin, A., Kirichek, R.: Testbed for identify IoT-devices based on digital object architecture. In: Galinina, O., Andreev, S., Balandin, S., Koucheryavy, Y. (eds.) *NEW2AN ruSMART 2018*. LNCS, vol. 11118, pp. 129–137. Springer, Cham (2018)
10. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system. *Decentralized Bus. Rev.* 21260 (2008)
11. Mohanta, B.K., Panda, S.S., Jena, D.: An overview of smart contract and use cases in blockchain technology. In: *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1–4. IEEE (2018)
12. Amanlou, S., Hasan, M.K., Bakar, K.A.A.: Lightweight and secure authentication scheme for IoT network based on publish–subscribe fog computing model. *Comput. Netw.* **199**, 108465 (2021)
13. Li, D., Peng, W., Deng, W., Gai, F.: A blockchain-based authentication and security mechanism for IoT. In: *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, pp. 1–6 (2018)
14. Almadhoun, R., Kadadha, M., Alhemeiri, M., Alshehhi, M., Salah, K.: A user authentication scheme of IoT devices using blockchain-enabled fog nodes. In: *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*, pp. 1–8 (2018)
15. Yavari, M., Safkhani, M., Kumari, S., Kumar, S., Chen, C.M.: An improved blockchain-based authentication protocol for IoT network management. *Secur. Commun. Netw.* **2020**, 1–16 (2020)
16. Kabir, R., Hasan, A.T., Islam, M.R., Watanobe, Y.: A blockchain-based approach to secure cloud connected IoT devices. In: *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, pp. 366–370. IEEE (2021)
17. Yang, X., Yang, X., Yi, X., et al.: Blockchain-based secure and lightweight authentication for internet of things. *IEEE Internet Things J.* **9**(5), 3321–3332 (2021)
18. Gong-Guo Z., Wan Z.: Blockchain-based IoT security authentication system. In: *2021 International Conference on Computer, Blockchain and Financial Development (CBFD)*, pp. 415–418. IEEE (2021)