J Grid Computing (2023) 21:69
https://doi.org/10.1007/s10723-023-09699-2

RESEARCH

Preservation of Sensitive Data Using Multi-Level Blockchain-based Secured Framework for Edge Network Devices

Charu Awasthi · Prashant Kumar Mishra · Pawan Kumar Pal · Surbhi Bhatia Khan · Ambuj Kumar Agarwal · Thippa Reddy Gadekallu · Areej A. Malibari

Received: 23 May 2023 / Accepted: 10 October 2023 / Published online: 17 November 2023

Abstract The proliferation of IoT devices has influenced end users in several aspects. Yottabytes (YB) of information are being produced in the IoT environs because of the ever-increasing utilization capacity of the Internet. Since sensitive information, as well as privacy problems, always seem to be an unsolved problem, even with best-in-class in-formation governance standards, it is difficult to bolster defensive security capabilities. Secure data sharing across disparate systems is made possible by blockchain technology, which operates on a decentralized computing paradigm. In the ever-changing IoT environments, blockchain technology provides irreversibility (immutability) usage across a wide range of services and use cases.

C. Awasthi
JSS Academy of Technical Education, Noida, India
e-mail: charuawasthi@gmail.com

P. K. Mishra
Department of Computer Science and Engineering, Pranveer Singh Institute of Technology, Kanpur, India
e-mail: prkm.cse@gmail.com

P. K. Pal
Department of Computer Science, KIET Group of Institutions, Ghaziabad, India
e-mail: pawan.pal@kiet.edu

S. B. Khan (✉)
Department of Data Science, School of Science, Engineering and Environment, University of Salford, Salford, UK
e-mail: s.khan138@salford.ac.uk; surbhibhatia1988@yahoo.com

S. B. Khan · T. R. Gadekallu
Department of Electrical and Computer Engineering, Lebanese American University, Byblos, Lebanon
e-mail: thippareddy.g@vit.ac.in

A. K. Agarwal
Department of Computer Science and Engineering, Sharda School of Engineering and Technology, Sharda University, Greater Noida, India
e-mail: Ambuj4u@gmail.com

T. R. Gadekallu
Zhongda Group, Haiyan County, Jiaxing City 314312, Zhejiang Province, China

T. R. Gadekallu
School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, India

T. R. Gadekallu
College of Information Science and Engineering, Jiaxing University, Jiaxing 314001, China

T. R. Gadekallu
Division of Research and Development, Lovely Professional University, Phagwara, India

A. A. Malibari
Department of Industrial and Systems Engineering, College of Engineering, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, 11671 Riyadh, Saudi Arabia
e-mail: aamalibari@pnu.edu.sa

Therefore, blockchain technology can be leveraged to securely hold private information, even in the dynamicity context of the IoT. However, as the rate of change in IoT networks accelerates, every potential weak point in the system is exposed, making it more challenging to keep sensitive data se-cure. In this study, we adopted a Multi-level Blockchain-based Secured Framework (M-BSF) to provide multi-level protection for sensitive data in the face of threats to IoT-based networking systems. The envisioned M-BSF framework incorporates edge-level, fog-level, and cloud-level security. At edge- and fog-level security, baby kyber and scaling kyber cryptosystems are applied to ensure data preservation. Kyber is a cryptosystem scheme that adopts public-key encryption and private-key decryption processes. Each block of the blockchain uses the cloud-based Argon-2di hashing method for cloud-level data storage, providing the highest level of confidentiality. Argon-2di is a stable hashing algorithm that uses a hybrid approach to access the memory that relied on dependent and independent memory features. Based on the attack-resistant rate (>96%), computational cost (in time), and other main metrics, the proposed M-BSF security architecture appears to be an acceptable alternative to the current methodologies.

Keywords Attack · Data protection · Cryptosystem · Encryption · Decryption · Hashing · Resistant · Overheads · Multiple securities · Computation

Abbreviations

ARR	Attack Resistant Rate
CPU	Central Processing Unit
DBT	Data Breaches and Theft
DDoS	Distributed Denial of Service
DNS	Domain Name System
DNS	Domain Name System
DoS	Denial-of-Service
DPI	Data Preservation Index
GHz	Gigahertz
ID	Identification
IIoT	Industrial Internet of Things
IoT	Internet of Things
KB	Kilobytes
KDF	Key Derivation Function
LAN	Local Area Network
LWE	Learning With Errors
MB	Megabytes
M-BSF	Multi-level Blockchain-based Secured Framework
MM	Man-in-the-Middle
NAB	Numenta Anomaly Benchmark
ODN	OriginTrail Decentralized Network
PII	Personal Identifiable Information
RFID	Radio Frequency Identification
SHA	Secure Hash Algorithm
SQL	Structured Query Language
UTC	Universal Time Coordinated
VM	Virtual Machine
VMH	Virtual Machine Hopping
YAML	Yet Another Markup Language
YB	Yottabytes

1 Introduction

In the Internet of Things (IoT) context [1–3], "dynamicity in IoT infra" is defined as the capacity of an IoT infrastructure to change and evolve in response to external stimulants. The IoT refers to a network of interoperable electronic devices that may exchange data with one another, along with cloud-based connectivity to facilitate several valuable functions. As a result of its adaptability and scalability, IoT infrastructures can accommodate a wide range of endpoints, sensors, and software. This adaptability is crucial in IoT settings, where the equipment and data sources are subject to regular change. Key aspects that allow flexibility in IoT systems include:

- For the IoT to be successful, its underlying technical infrastructure must be extensible enough to accommodate an ever-increasing number of connected devices and data sources.
- IoT backbones must be compatible with a wide range of devices and data sources, irrespective of brand or communication protocol.
- The network structure, equipment accessibility, and information quantity may all undergo changes, and IoT infrastructures must be flexible enough to handle them.
- Infrastructures for the IoT need to be flexible enough to adapt rapidly to new circumstances, such as introducing new services and programs.

In general, the dynamism of IoT infrastructures is crucial, as it allows them to accommodate the

ever-increasing complexity of the IoT ecosystem [4].In the IoT domain, "sensitive data" refers to any information that could be used to uniquely identify a person or reveal private information (like PII, financial data, etc.). IoT devices create an ever-increasing amount of data, and it's becoming more critical to preserve this data so it doesn't get lost, stolen, or corrupted [5].

1.1 Necessity of Preservation of Sensitive Data

Preserving data in the IoT entails taking precautions to prevent the loss, corruption, or manipulation of data and keeping it accessible for as long as possible [6]. Information needs to be protected and recovered whenever accessed by its intended users to prevent unwanted parties from having access to it. To secure the long-term preservation and protection of sensitive data in the IoT, a complex strategy, including technological solutions, rules, and procedures, is required [7, 8]. Several IoT applications, including those in financial services, healthcare, and critical facilities, rely on sensitive data for proper functioning [9, 10]. Losing this data can have severe consequences for the functioning of these applications, with adverse economic and social effects.

1.2 Significance of Cryptosystems and Blockchain Technology

Blockchain and other robust cryptosystems could help a loT with the safe storage of private information in IoT networks [11]. The immutability and inalterability of blockchain-created records guarantee the truthfulness and reliability of any information stored within. Furthermore, since each document is cryptographically connected to its predecessor, any tampering with the data will be immediately detected. This safety measure prevents unauthorized changes to private information like bank account numbers or social security numbers. To ensure the safety of sensitive data in IoT infrastructure, blockchain technology and cryptosystems may provide additional layers of protection. Using such tools allows any business (that relies on IoT infra) to protect its customers' personal information, comply with laws, and keep their confidence [12]. Blockchain and cryptosystems can be used at the edge device, in thefog, and in the cloud, among other places in the IoT architecture, to make IoT services more secure and less centralized.

1.3 Limitation and Utilization

There are benefits to using blockchain and cryptosystems for data preservation in the IoT, but there are also limitations. Figure 1 shows some of the limitations and ways that blockchain and cryptosystems can be used to protect data in the IoT domain.

Limitation Large-scale IoT applications might be challenging to scale due to blockchain technology's slowness and resource requirements. This may restrict its use in IoT systems that produce copious amounts

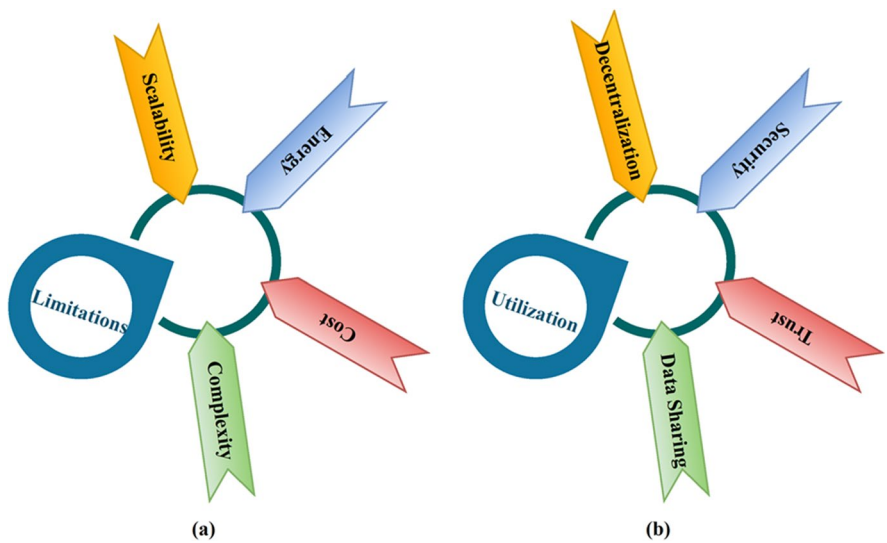

Fig. 1 (**a**) Limitation and (**b**) Utilization of Blockchain and Cryptosystem in Data Security Process

of data. Power-constrained IoT devices may be unable to keep up with the high energy demands of blockchain processing and transaction validation. Costs associated with implementing and maintaining blockchain technology may prevent its widespread use in IoT systems. For less technically savvy IoT devices, blockchain deployment and integration complexity may prevent its use.

Utilization Decentralized data management, made possible by blockchain and cryptosystems, has the potential to improve both security and openness. Cryptography is used by blockchain and other cryptosystems to ensure the safety of data and transactions, making them more resistant to hacking and other forms of cybercrime [13]. The immutability and transparency of blockchain and cryptosystems may increase trust between participants in IoT applications. Blockchain and other cryptosystems may make it easier for many parties to share data in IoT applications in a safe and transparent way, which in turn can boost cooperation and creativity.

1.4 Scope of the Research Work

The scope of research work for the preservation of sensitive data using multi-level blockchain-based secured framework for dynamic IoT infrastructure is extensive and can cover several areas of research. This study field concentrates on the creation of blockchain-based solutions for the storage of data in IoT architecture. Protecting private information [14] in IoT infrastructure is the primary goal of this research work. To prevent illegal access and maintain data privacy, measures such as encryption and intrusion detection must be implemented. In addition, research in this field may focus on finding ways to accommodate the ever-changing requirements of IoT infrastructure. Improving the availability and dependability of IoT systems requires the development of self-healing systems, reliable architectures, and highly adaptable techniques. The effectiveness of multi-level blockchain-based protected frameworks for pliable IoT architecture can potentially be studied in this research. Assessing the practicability and efficiency of these solutions requires looking at things like ARR, DPI, computational overheads, and storage overheads.

1.5 Motivation

The motivation behind the research work on "preservation of sensitive data using multi-level blockchain-based secured framework for dynamic IoT infrastructure" is to address the critical challenge of preserving sensitive data in dynamic IoT environments. The confidentiality and safety of sensitive data have evolved into a significant issue for businesses across a wide range of sectors due to the expansion of IoT devices and the rising volume of data produced by those devices [15].

Centralized databases and encryption, two standard methods for preserving data in IoT infrastructure, have their limits when protecting sensitive information. Blockchain technology's decentralized and transparent way of data storage may provide high levels of protection for users' personal information and other sensitive data [16]. However, several obstacles to overcome when integrating blockchain technology into IoT infrastructure include expansion, energy consumption, and intricacy.This study aims to create a multi-level blockchain-based protected architecture that can deal with these issues and provide practical solutions for protecting sensitive data in the ever-changing IoT environment. The suggested framework seeks to use solid cryptosystem processes and blockchain technology's privacy and security capabilities while also resolving issues with expansion, energy consumption, and complexity. Furthermore, the suggested architecture uses a multi-level security strategy to guarantee that sensitive data is safeguarded all the way through the IoT infrastructure, from the edge device to the cloud.

1.6 Objectives

The major objective of the research are delineated as follows.

- To create a multi-level blockchain-based protected framework by integrating strong cryptosystems for protecting private data in an ever-changing IoT environment.
- To evaluate the effectiveness and feasibility of the proposed multi-level blockchain-based secured framework. Computational overheads, storage overheads, ARR, and DPI, may all be assessed by putting the framework through its paces in a

simulated IoT infrastructure. The analysis aims to prove that the suggested framework can effectively secure private information in an evolving IoT setting. Ultimately, this research aims to provide a complete framework for protecting private data in dynamic IoT infrastructure, therefore contributing to the development of IoT security and privacy. As a result, the study's results could be used to help users and other interested parties have more faith in IoT systems.

- The delineation design of this article is organized as follows: Section 2 briefs the conceptual techniques of most recent existing works; Section 3 elaborates the proposed security framework for dynamic IoT infrastructure; Section 4 discusses the evaluation metrics and comparative analysis based on the attained outcomes; and Section 5 wraps up the research with vital conclusive points, highlighting the significance of this research work and future enhancements.

2 Related Work

This section talks about the different ways that blockchain-based IoT systems handle authentication, sharing data, and getting data back while keeping privacy safe. In addition, we investigate the various blockchain architectural frameworks available to support multiple IoT applications.

[17] is still a unique blockchain network built specifically for the IoT. The platform was so-called to honor Charles Walton, the man responsible for developing RFID technology, and to recognize his efforts in furthering his dream of RFID's widespread adoption via the IoT. Many distinct blockchains may coexist on the Waltonchain platform. The Waltonchain, which serves as the foundation for all other blockchains, is a public ledge devicer that enables any users who meet specific requirements to participate in its consensus process and perform various roles inside the blockchain. The characteristics of this public blockchain remain consistent with those of previous public blockchains. Waltonchain's goal is to disrupt present IoT businesses by combining the blockchain's credibility, auditability, and traceability with RFID in IoT devices. One potential drawback of the Waltonchain concept is that it might require a substantial investment in time, money, and other resources to deploy well. Sophisticated equipment, code, and infrastructure elements may be needed for the framework's intricate integration of RFID and blockchain technology. Ultimately, it would be hard for smaller firms and organizations with fewer resources to use the technology. Also, security and regulatory issues should be carefully evaluated before deploying the framework, as is the case with any cutting-edge technology.

NetObjex [18] is a platform for the decentralized management of digital resources that have solutions for supply chain operations, the processing industry, the industrial internet, and industrial machinery. The framework includes several communication formats and incorporates the IoT and blockchain to collect and disseminate data. The intricacy and steep learning [19, 20] curve of the NetObjex framework might be seen as a drawback. It may take some time for designers/developers to gain an understanding of making the most of all of its features because of its wide range of capabilities and support for different platforms and protocols. Substantial time and skill may also be required to integrate the framework with preexisting systems.

Moeco is a blockchain platform designed to become the "DNS of things" [21]. There are billions of things in the world that can now connect to the internet, attributable to this platform and its implementation of many network protocols and associated gateways. Specifically designed to work with IoT devices, it also manages to keep costs down. However, the centralized nature of the Moeco framework is a possible drawback that might lead to information security and privacy issues. Hence, all information is stored and processed on Moeco's servers, which may be susceptible to hackers and other forms of data breaches. The need for centralization may also hinder the framework's extensibility since it may become increasingly difficult to manage massive volumes of data as the range of networked devices expands. Lastly, regulatory issues should be carefully evaluated before applying the framework, as they could be present with any new technology.

OriginTrail is a permissionless blockchain technology that allows distributed data exchange between several organizations [22]. This system ensures data integrity by combining a distributed ledger with automated supply chain operations. The central concept is to use a uniform blockchain-based approach via an

incentive-based system to guarantee product objectives and customer safety. OriginTrail resolves two of the most important causes of disruptions in the supply chain's data collection and sharing processes. On top of the blockchain layer, the system's infrastructure and analysis activities construct a distributed peer-to-peer platform called ODN. The OriginTrail architecture has the potential drawback of requiring specialized knowledge and comprehension of blockchain technology for optimal performance. This may be difficult for those without technical expertise in blockchain or the means to employ dedicated programmers. Also, privacy and regulatory issues should be carefully evaluated before deploying the framework, as with any cutting-edge technology.

This [23] gave three possible designs for blockchain systems based on how IoT devices and faraway blockchain clients communicate with each other. The goal is to cut down on traffic even more, and blockchain is thought to be able to help with low-bitrates and low-power mobile technology. In the present context, this assumption isn't true in the IIoT and only applies to empirical testing. [24] used blockchain solutions to build a generic IoT gateway that can manage a large number of IoT devices efficiently. The blockchain-based, revolutionary IoT back-end technology has been developed to be resilient to DDOS assaults on a decentralized storage network. [25] suggested using a blockchain-based infrastructure to facilitate firmware updates for IoT devices. This platform not only guarantees the confidentiality of endpoint device firmware but also verifies and authenticates firmware updates from device makers.

This [26] made a new internet-based incentive framework based on the incentive principle of distributed ledger technology to encourage and reward participants for backtracking intrusion detection mechanisms and sharing detection data. The method allows platform users to proactively sustain the network's regular functioning in exchange for benefits, much like Bitcoin mining. [27] suggested a traceable framework system to tackle the quality difficulties of agricultural goods supplied to the market. This architecture can track a variety of data, including the source and supplier of a specific resource, in real-time. This architecture was developed by employing blockchain solutions for tracking.

Some possible research gaps in the preservation of sensitive data using blockchain-based multi-level secured framework for dynamic IoT infrastructure are lack of empirical studies, scalability, and integration with existing model, preservation and adoption. Thus, we focus on applying multi-level security process at different level of IoT platform.

3 Multi-Level Blockchain-Based Secured Framework

A multi-level blockchain-based secured framework is a security architecture that uses blockchain technology with robust cryptosystems to create a secure and tamper-proof environment for data and transactions. This framework utilizes a multi-level approach to security, with multiple layers of protection to ensure data privacy and confidentiality. By leveraging the immutability and transparency of blockchain, this framework provides a robust and trustworthy solution for a wide range of applications, including finance, healthcare, and supply chain management. Figure 2 represents the overall proposed architecture of M-BSF. The architectural design depicts that the data protection and secure transaction is ensured at all the three levels of IoT infra (Edge-level, Fog-level, and Storage-level at cloud).

3.1 Datasets

To validate any data protection model, it is essential to evaluate the privacy and protection processes through the deployment of standard datasets. Such datasets not only highlights the strengths of the model but also exposes the loopholes against the complicated attacks. Thus, concerning all the required facts, we incorporated NAB dataset [28]. The NAB dataset contains a collection of real-world time-series data that are suitable for anomaly detection. It includes data from a wide range of sources, including sensors, servers, and social media. The dataset is designed to be challenging, with a high degree of variability and complexity, making it a good choice for testing your model in a dynamic infrastructure. The NAB dataset includes both normal and anomalous data, allowing the researchers to validate the proposed model's ability against the detection of anomalies in dynamic IoT infrastructure. It contains a variety of datasets, which is highlighted in the Table 1.

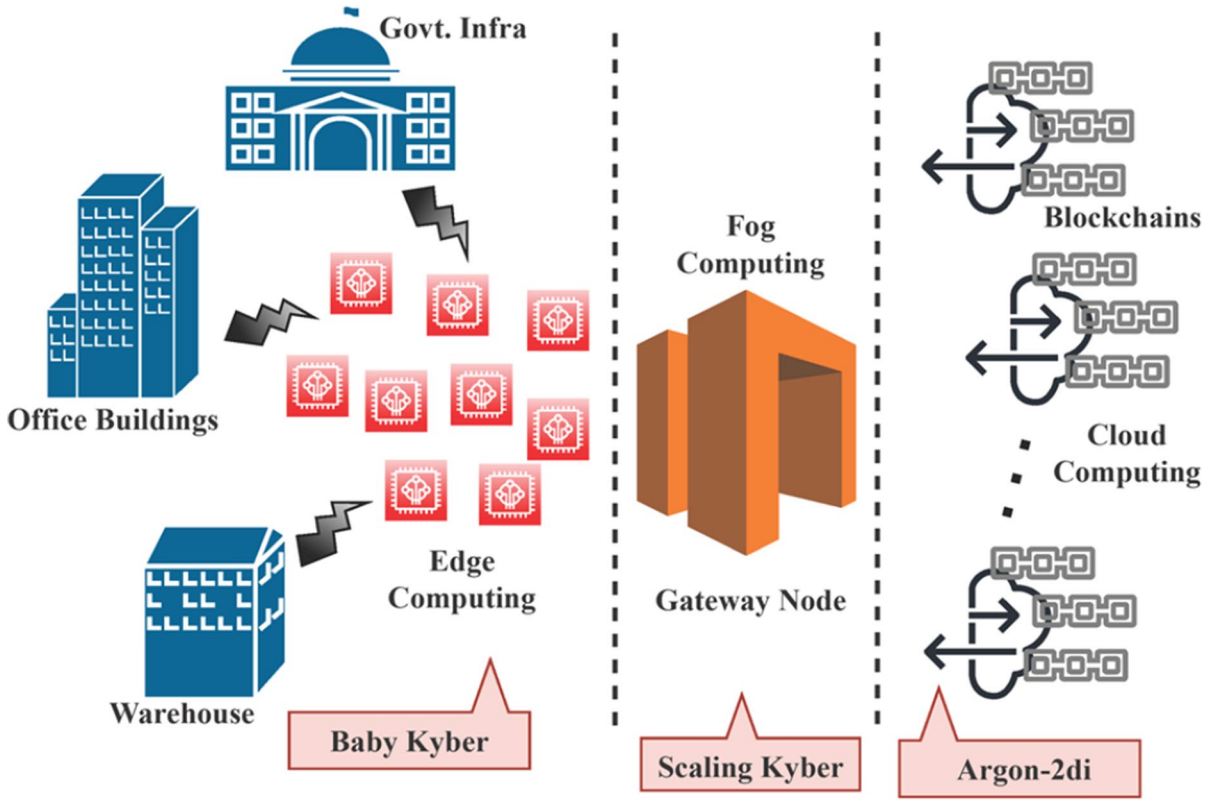

Fig. 2 Architecture of M-BSF

Table 1 Dataset Collection of NAB and its Vital Attributes

Dataset Name	Description	Attributes
Machine Temperature	Temperature readings from a machine used to manufacture computer chips	Time series data, sensor readings, temperature
Ambient Temperature	Temperature readings from sensors deployed in an office environment	Time series data, sensor readings, temperature
Traffic	Traffic data from sensors deployed on a major freeway	Time series data, sensor readings, traffic volume
Power Consumption	Power consumption data from smart meters	Time series data, sensor readings, power consumption
CPU Utilization	CPU utilization data from servers	Time series data, server data, CPU utilization
Network Traffic	Network traffic data from a LAN	Time series data, network data, traffic volume
Anomaly Ground Truth	Contains labeled anomalies for the other datasets	Labeled data, anomaly detection, ground truth

Each of these datasets includes time series data, which is common in IoT-based dynamic infrastructures where data is collected over time. In addition, each dataset includes sensor readings or other data from IoT devices or other sources, and is designed to be challenging with a high degree of variability and complexity to simulate real-world scenarios. Since NAB dataset does not include blockchain-oriented data, it is essential to integrate another dataset that highly support the testing of Kyber-based

blockchain model. Thus, we incorporated BigQuery [29], which is a public dataset of the Ethereum blockchain, a fully-managed, serverless data warehouse. This dataset includes the entire Ethereum blockchain history, including all transactions, addresses, and contracts.

The dataset could also include data related to the IoT devices that are interacting with the blockchain network. This could include information on the type of device, the frequency of interactions, and the amount of data being transmitted. By combining this data with blockchain performance data, it may be possible to identify patterns and trends that could be used to optimize the performance of the network. Some possible attributes of a BigQuery dataset with resource-based instance is represented in Table 2, which may differ based on the concerned evaluation context.

3.2 Edge-Level Security Process

With dynamic IoT infrastructures, edge-level security procedures are very important for making sure that the whole system is safe and secure. Edge computing includes processing data at the edge of the network, relatively close to the originator of the information, rather than transmitting all the data to a centralized facility for processing. Even though this method improves speed and reduces latency, it also raises concerns about the security risks of IoT devices [30–34]. Edge-level security procedures are meant to protect IoT devices from hacking, malware, and data breaches, among other security risks. Some of the primary advantages of this form of security procedure include decreased latency, increased data privacy, data scalability, and so on. Securing IoT device-to-device communication in dynamic IoT infrastructures is achievable through the use of the Baby Kyber post-quantum cryptographic algorithm. This cryptosystem is a quantum-safe variation of the Kyber one.The Baby Kyber cryptosystem makes use of a public key encryption approach that is predicated on the complexity of the LWE issue. This issue involves discovering a secret vector $\hat{s}$ from a collection of random linear functions of the type $\alpha \bullet \hat{s} + \varepsilon = \delta$, where α arbitrary matrix, δ is is a randomized vector, ε is a relatively small erroneous vector, and $(\cdot)$ signifies the dot product.

Sender-side encryption necessitates the generation of a random vector δ and the subsequent computation of a ciphertext $\mathbb{C}$, as shown below:

Table 2 Significant Attributes of BigQuery Dataset

Attribute Name	Description	Example Value
Transaction ID	A unique identifier for each transaction processed by the blockchain network	0×12ab34cd56ef78
Timestamp	The time at which each transaction was processed by the network	2023–03-14 14:36:42 UTC
Resource Used	The amount of computational resources used to process each transaction	21,000
Resource Price	The fee paid by the sender of each transaction to prioritize its processing on the network	50 Gwei
Sender Address	The address of the user or device that initiated each transaction	0×1234567890abcdef
Receiver Address	The address of the user or device that received each transaction	0×9876543210fedcba
Device ID	A unique identifier for each IoT device interacting with the blockchain network	IoT123
Device Type	The type or model of the IoT device	Raspberry Pi
Data Volume	The amount of data transmitted by each IoT device	500 KB
Network Latency	The time delay between sending and receiving data for each IoT device	50 ms
Block Number	The number of the block in which each transaction was included	1,234,567
Block Timestamp	The time at which each block was created by the blockchain network	2023–03-14 14:40:02 UTC
Resource Limit	The maximum amount of computational resources that can be used to process transactions in each block	12,500,000
ResourceUsed Percentage	The percentage of the resource limit used to process transactions in each block	0.16%

$$\mathbb{C} = [(\boldsymbol{\alpha} \cdot \boldsymbol{\delta}), (\boldsymbol{M} \cdot \boldsymbol{P}_{\mathbb{k}}) + (\boldsymbol{\alpha} \cdot \boldsymbol{S}_{\mathbb{k}}) + \boldsymbol{\varepsilon}] \quad (1)$$

where $P_{\mathbb{k}}$ is the public key, $S_{\mathbb{k}}$ is the secret key and M is the message to be encrypted. The sender then sends the $\mathbb{C}$ to the receiver.To decrypt the $\mathbb{C}$, the receiver computes:

$$\boldsymbol{M} = [(\mathbb{C}_2) - (\mathbb{C}_1 \cdot \boldsymbol{S}_{\mathbb{k}})] \quad (2)$$

where $\mathbb{C}_1 = (\boldsymbol{\alpha} \bullet \boldsymbol{\delta})$ and $\mathbb{C}_2 = [(\boldsymbol{M} \cdot \boldsymbol{P}_{\mathbb{k}}) + (\boldsymbol{\alpha} \cdot \boldsymbol{S}_{\mathbb{k}}) + \boldsymbol{\varepsilon}]$. The receiver can then recover the original message M. The Baby Kyber cryptosystem is resistant to side-channel attacks, which can be a concern in IoT environments where physical access to devices is not always restricted. Table 3 represents the algorithm that comprises the computational process of baby kyber at edge-level communication.

The Baby Kyber post-quantum cryptography algorithm offers several benefits in securing IoT device-to-device communication in dynamic IoT infrastructures. Some of the key advantages of the Baby Kyber algorithm are Quantum resistance, Public key encryption, Complexity based on LWE Data privacy, and Scalability.

3.3 Fog-Level Security Process

Kyber cryptosystem performance on edge devices with low resources may be improved by improving its implementation to decrease computational overhead and memory needs, which is a crucial part of scaling the system for use in fog-level computing. The security of fog-level computing systems may also be strengthened by scaling the Kyber cryptosystem. Even on edge devices that may be more susceptible to security breaches, essential data may be shielded from assaults and unauthorized access with the help of secure and effective cryptographic processes.Many considerations are necessary for scaling the Kyber cryptosystem in dynamic IoT infrastructures at the fog computing level. Some crucial factors include: Restricted resources, a rapidly evolving context, and safety issues, all need to be addressed. Many scientific

Table 3 Algorithm 1: Step-wise Algorithm of Baby Kyber in Dynamic IoT infra at Edge-Level Cryptosystem

Input: M, $\boldsymbol{P}_{\mathbb{k}}, \boldsymbol{S}_{\mathbb{k}}$ Output: $\mathbb{C}$, M	
Key Generation	1: Choose N → size(ƙ) 2: Choose a polynomial ring (я). $я = \boldsymbol{Q}(\boldsymbol{x}) \big/ (\boldsymbol{x}^N + \boldsymbol{1})$ (3) where, Q is a prime number such that q > (2n+1). 3: Choose $\boldsymbol{r_p}$ in я of degree less than N. //rp is random polynomial 4:Choose $\boldsymbol{r}_{\lvert m \rvert}$ //$r_{\lvert m \rvert}$is random matrix 5: Compute: $\boldsymbol{P}_{\mathbb{k}} = (\boldsymbol{r}_{\lvert \boldsymbol{M} \rvert} + \varepsilon)$ (4)
Encryption	6: Choose $\boldsymbol{r_p}$ in я of degree less than N. 7: Compute: $\mathbb{C} = [\mathbb{C}_1, \mathbb{C}_2] // \mathbb{C}_1 = \boldsymbol{r}_{\lvert \boldsymbol{m} \rvert}^{\boldsymbol{r_p}}, \mathbb{C}_1 = \boldsymbol{M} \cdot \boldsymbol{P}_{\mathbb{k}} + \boldsymbol{r} + \varepsilon$ (5)
Decryption	8: Compute the inner product of $\mathbb{C}_1$and s $\boldsymbol{\mu} = [\mathbb{C}_1 * \boldsymbol{s}]$ (6) 9: Compute ε as ε': $\boldsymbol{\varepsilon}' = [\mathbb{C}_2 - \boldsymbol{\mu}]$ (7) 10: Recover *M* $\boldsymbol{M} = \left\lceil \boldsymbol{Q}/\boldsymbol{2N} \cdot (\boldsymbol{\varepsilon}') \right\rfloor$ /round_to_nearest (8)

computations are needed to scale the Kyber cryptosystem at the fog computing level in dynamic IoT systems. To produce a working public–private key pair, the following calculations must be carried out, which is completely represented in algorithm 2 (Table 4).

3.3.1 Optimization of Scaling Kyber

In general, these mathematical processes need to be improved and parallelized so that the Kyber cryptosystem may be effective in fog computing settings and scalable to a greater degree. In addition, the Kyber cryptosystem may take advantage of batch processing to enhance the efficiency of the encryption and decryption procedures. Minimizing computational overhead and boosting system efficiency is feasible by encrypting or decrypting countless $\mathbb{C}$ simultaneously utilizing a single public or private key through batch processing. Table 5 depicts the technical description of the Kyber cryptosystem's batch processing procedure as follows. Table 5 denotes algorithm of kyber cryptosystem's batch processing.

3.4 Cloud-Level Security Process

The blockchain technology is often employed for secure and decentralized data storage. To store data on the blockchain, the data needs to be converted into a unique and secure hash that can be easily verified but cannot be easily reversed. This is where the Argon-2di hashing method can be useful. Argon-2di algorithm is specifically designed for applications where the attacker can perform arbitrary memory access. When using the Argon-2di hashing method for cloud-level data storage in blockchain, the following steps are needed to be executed. Data is first converted into a binary format, which is then divided into fixed-size chunks for efficient processing.Each chunk is then hashed using the Argon-2di hashing method

Table 4 Algorithm 2: Step-wise Algorithm of Scaling Kyber in Dynamic IoT infra at Fog-Level

Input: n, P *//n is a positive integers, P is the prime power of 2*	
Key Generation	1: Generate S; $\boldsymbol{S} \in \boldsymbol{\varphi_Q^n}$,where S is a random n-dimensional vector over φ_Q. 2: Compute the matrix 'm' as the transpose of the$P_k(x)$, where $P_k(x)$ is generated by sampling n polynomials Pi(x) of degree at most d with coefficients inφ_Q. 3: Compute: $\boldsymbol{P}_k(\boldsymbol{x}) = (\boldsymbol{P_i}, \chi)$ (9) where, χ is the result of the matrix-vector multiplication of $\boldsymbol{P_i}$ and S. 4: Compute: $\xi_k \leftarrow S$
Encryption	1: Generate U; $\boldsymbol{U} \in \boldsymbol{\varphi_Q^n}$, where U is a random n-dimensional vector over φ_Q. 2: Compute the matrix 'n' as the transpose of the$P_k(x)$, where $P_k(x)$ is generated by sampling n polynomials Vi(x) of degree at most d with coefficients inφ_Q. 3: Compute: $\mathbb{C} = (\boldsymbol{V}, \boldsymbol{o})$ (10) where 'o' is the outcome of the of V and r, added to the M encrypted under a short-term symmetric key. 4: Compute: $\boldsymbol{SS}_k$ and $\boldsymbol{\psi}_k$ //SS denotes shared secret key and ψ indicates shared symmetric key $\boldsymbol{\psi}_k = \boldsymbol{KDF}(\boldsymbol{SS}_k \| \boldsymbol{\eta})$ (11) where $\psi_k$is computed using KDF with inputs SS_k and a nonce η. 5. $\mathbb{C} = \boldsymbol{encrypt}(\boldsymbol{\psi}_k, \boldsymbol{M})$ (12)
Decryption	6: Compute $\boldsymbol{SS}_k$ and $\boldsymbol{\psi}_k$ 7: $\boldsymbol{M} = \boldsymbol{decrypt}(\mathbb{C}, \boldsymbol{\psi}_k)$ (13)

Table 5 Algoirthm 3: Batch Processing of Kyber Cryptosystem

Input: $M = \{m_1, m_2, \cdots, m_k\}$	
Batch Encryption	1: Generate m_k randomized n-dimensional vector $\{\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_k\}$ over φ_Q 2: Compute the matrix 'n' as the transpose of the$P_{\mathrm{k}}(x)$, where $P_{\mathrm{k}}(x)$ is generated by sampling n polynomials Bi(x) of degree at most d with coefficients inφ_Q. 3: Compute $\bar{x}_k \rightarrow \{\mathbb{C}_1, \mathbb{C}_2, \cdots, \mathbb{C}_k\}$ as FOR i from 1→k Compute Ui //as the outcome of the 'n' and r, added to the M encrypted under a $\boldsymbol{\psi}_{\mathrm{k}}$. $\boldsymbol{U_i} = (\boldsymbol{n} * \bar{\boldsymbol{x}}_{\boldsymbol{k}}) + \boldsymbol{e}[\boldsymbol{m_k}]$ (14) $\boldsymbol{U_i} = \{\boldsymbol{U_1}, \boldsymbol{U_2}, \cdots, \boldsymbol{U_k}\}$ (15) 4: Compute SS_{k}; $\bar{\boldsymbol{S}} \sim \boldsymbol{\varphi_{Q^n}}$ (16) $\boldsymbol{SS}_{\mathrm{k}} = \boldsymbol{P}_{\mathrm{k}}(\boldsymbol{x}) * \bar{\boldsymbol{S}}$ (17) 5: Compute ψ_{k}; $\boldsymbol{\psi}_{\mathrm{k}} = \boldsymbol{KDF}(\boldsymbol{SS}_{\mathrm{k}} \| \boldsymbol{\eta})$ (18) $\boldsymbol{V} = \{\boldsymbol{m_1}, \boldsymbol{m_2}, \cdots, \boldsymbol{m_k}\}$ (19) 6: Compute: $\mathbb{C}_{\boldsymbol{k}} = (\boldsymbol{U}, \boldsymbol{V})$ (20)
Batch Decryption	7: Split $\mathbb{C}_{\boldsymbol{k}} \rightarrow (\boldsymbol{U}, \boldsymbol{V})$ $\mathbb{C}_{\boldsymbol{k}} = (\boldsymbol{U_i} \| \boldsymbol{V_i})$ (21) 8: Compute SS_{k}; 9: Compute $\boldsymbol{\psi}_{\mathrm{k}} = \boldsymbol{KDF}(\boldsymbol{SS}_{\mathrm{k}} \| \boldsymbol{\eta})$ 10: $\forall(V_i)$ do $\boldsymbol{m_k} = \boldsymbol{D}[\boldsymbol{\psi}_{\mathrm{k}}, \boldsymbol{V_i}]$ (22)

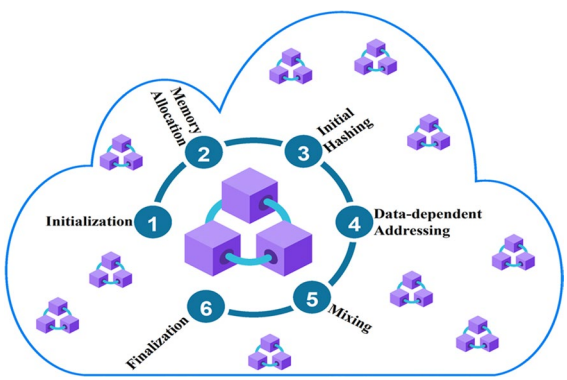

Fig. 3 Cloud-Level Security Process Flow

with a unique salt value. The resulting hash values are combined using a secure and irreversible function such as Blake2b or SHA-256 to generate a single hash value. The final hash values are then stored on the blockchain as a unique and secure representation of the original data. Figure 3 represents the cloud-level security process flow in a diagrammatical way.

The Argon-2di hashing function is a complex algorithm that involves several steps and computations, but here are the mathematical equations that describe the basic steps:

Initialization: The password or key to be hashed is one input, and other factors such as memory size, number of iterations, and number of threads are also required. Let ρ be the password or key to be hashed, and let ϒ be a random salt value of size s.

Memory allocation: Assigned memory is filled with a "salt," a random value generated by the method.Let ϻ be a memory block of size b, $b \geq 2 * s$.

Initial hashing: At the first stage of the process, the password or key is hashed with the salt as well as other parameters as inputs.

$$\boldsymbol{f}(\mathrm{h}) = \mathrm{h}[\boldsymbol{\rho} \| \mathbf{s} \| \mathbf{p}] \tag{23}$$

where, $\boldsymbol{f}(\mathrm{h})$ denotes hash function and Blake2b hash function is utilized to compute $\boldsymbol{f}(\mathrm{h})$.

Data-dependent addressing: The method makes advantage of data-dependent addressing to perform memory access in a manner that is specific to the password or key is hashed. Using this method, it will be more challenging for attackers to execute time-memory trade-off assaults.

$\forall(i = 0 \rightarrow t-1)$*//t indicates number of iterations*
$\mathrm{M} \rightarrow \{\mathrm{M}_0, \mathrm{M}_1, \cdots, \mathrm{M}_{(t-1)}\}$
$\forall(j = 0 \rightarrow z-1)$*//z indicates number of lanes*
Compute: i, j, x, y, w

$$i = 2 * i + 1;$$
$$j = i \,\%\, b;$$
$$x = \mathrm{h}(\mathrm{i}||\mathrm{h}0)$$
$$y = x \,\%\, t;$$
$$z = x/t;$$

Compute the addresses a and b, as follows:

$$a = [(2 * j + 1) * z] \,\%\, b$$
$$b = [(2 * j + 2) * z] \,\%\, b$$

Retrieve the data D[a] and D[b] from the memory block M, and || them with x and y, respectively. Store the result back in M [a] and M [b].

Mixing: Using a mixture of bitwise operations and memory accesses, the method executes a series of mixing operations on the data in the memory block. This mixing procedure was developed to provide even higher defenses against time-memory trade-off assaults.

Finalization: The algorithm performs a final round of hashing on the data in the memory block, using the salt and other parameters as inputs. The resulting hash value is the output of the Argon2di function, which can be represented mathematically as:

$$\boldsymbol{A}rgon2di(\rho, \Upsilon, m, t, z) = f(\mathrm{h}) = \mathrm{h}(\mathrm{h}0 \,||\, \mathrm{M}) \quad (24)$$

3.5 Blockchain Integration

When integrating Argon-2di hashing into blockchain architecture, the primary consideration is the block's structure. Each block contains data, the hash of this data (using Argon-2di), and the hash of the previous block, creating a cryptographic chain. The Argon-2di hash ensures that even a minuscule change in data leads to a vastly different hash, making unauthorized alterations easily detectable. When a new block is added, nodes in the blockchain network employ a consensus mechanism, like Proof-of-Work, where Argon-2di can be utilized to compute the required cryptographic puzzle. This ensures that the added block's data is valid. Moreover, for block validation, each node recalculates the Argon-2di hash of the block's data and checks it against the block's stored hash. The resilience of Argon-2di against time-memory trade-off attacks further solidifies the blockchain's security, making it harder for malicious entities to alter past transactions or add fraudulent

$\forall(j = 0 \rightarrow t-1)$*//t indicates number of iterations*

$$\mathrm{M} \rightarrow \{\mathrm{M}_0, \mathrm{M}_1, \cdots, \mathrm{M}_{(t-1)}\}$$
$$\forall(k = 0 \rightarrow z-1)$$

Compute: i, j, x, y, w

$$i = 2 * i + 1;$$
$$j = i \,\%\, b;$$
$$x = \mathrm{M}[(\mathrm{j} + \mathrm{t} - 1) \,\%\, \mathrm{t}][(\mathrm{k} + \mathrm{z} - 1) \,\%\, \mathrm{z}]$$
$$y = \mathrm{M}[\mathrm{j}][\mathrm{k}]$$
$$z = \mathrm{M}[(\mathrm{j} + 1)\% \, t][(\mathrm{k} + 1) \,\%\, z]$$

$R = f_{\boldsymbol{\varpi}}(x = y + z)$*//*$f_{\boldsymbol{\varpi}}$*denote the mixing function*
Store the outcome R in M[j][k].

blocks. Integrating such a robust hashing mechanism directly ties into the core principles of blockchain: decentralization, transparency, and immutability.

A simple consensus mechanism called "Proof-of-Elapsed Time (PoET)" is incorporated with the Argon-2di hashing. The PoET mechanism is especially suitable because it is less energy-intensive than Proof-of-Work, yet it ensures fairness in the decentralized network.

Proof-of-Elapsed Time (PoET) with Argon-2di integration:

Initialization Every participating node in the network is required to generate a random wait time using

a secure function. For our case, we'll use the Argon-2di hashing function to generate this wait time. The node hashes its unique ID (or other relevant data) using Argon-2di to produce a hash value, which is then converted to a numerical value representing the wait time.

Waiting Period Each node waits for its respective wait time. The idea is that the first node to complete its wait time gets the right to create the next block.

Block Creation Once a node's wait time elapses, it broadcasts its intention to create the next block. Along with this broadcast, it sends its Argon-2di generated hash (from step 1) as proof that it waited the intended time.

Validation Other nodes in the network validate the claim by:

- Recomputing the Argon-2di hash of the claiming node's unique ID to ensure the wait time is valid.
- Checking the block's content for validity.

Block Addition If a majority of nodes agree on the validity of the new block (both in terms of wait time and block content), it's added to the blockchain.

The integration of Argon-2di into PoET ensures that the wait time generation is both secure and unpredictable. This mechanism, while simple, leverages the security of Argon-2di while ensuring fairness and energy efficiency in block generation.

4 Performance Analysis

4.1 Experimental Setup

Four common blockchain frameworks [3] are evaluated at the experimental research level to assess the efficacy of the suggested M-BSF. First, an empirical testing bed is based atop Hyperledger Caliper. This is accomplished using Kubernetes version 1.24.11; Kubernetes YAML files version 1.0, and a node description version 16.13.1. Early on, 1000 peers from 5 enterprise domains are used to estimate the average, maximum, and minimum latency of performing the request operation and establishing a transaction in each comparative framework. A dual-core VM handled exceptional network processing with four gigabytes of memory running on a host Intel i7-7700 CPU (8 MB cache, up to 4.20 GHz). Two datasets, namely NAB and BigQuery, which were already elaborated on in Section 3.1, are considered to validate the proposed M-BSF.

4.2 Evaluation Metrics and Analysis

Metrics for evaluating a model's efficacy are helpful for making informed comparisons and selections among available options. In addition, they help pinpoint problem areas so that the suggested models may be fine-tuned for optimal performance. Few vital metrics are incorporated to evaluate the models. They are attack resistant rate, computation overheads, and storage overheads, and data preservation index.

4.2.1 ARR

An intruder attack resistant rate (ϕ) in an IoT environment can be computed as follows.

$$\Phi = [1 - \mathbb{P}] * 100 \tag{25}$$

where, $\mathbb{P}$ represents the probability of a successful intruder attack. In other words, the rate would represent the percentage of intruder attacks that are prevented by the security measures in place. Moreover, such resistant rate is measured against various attack types at different computing stages of IoT infra, which are listed in Table 6.

Figure 4 compares the ARR of various blockchain-based frameworks, including M-BSF, Waltonchain, NetObjex, Moeco, and OriginTrail, to different types of attacks, such as malware injection attacks, side-channel attacks, and DoS attacks. The result observed in Fig. 4(a) exposes the essential aspect of ensuring the security and privacy of data at the edge computing level. In dynamic IoT infrastructures, where devices are constantly added or removed, edge computing security becomes even more critical. Based on the outcome, we can observe that M-BSF has the highest ARR to malware injection, DoS, and side-channel attacks, which is almost 98%. On the other hand, OriginTrail, and NetObjex have the lowest ARR (lowest susceptibility) against malware injection, and side-channel attacks, respectively. Though

Table 6 Various Attack Types and its Descriptions

	Attack Type	Description
Edge computing	a. Malware injection attacks	The attacker injects malware into the edge device or sensor, which can cause it to malfunction, compromise sensitive data, or participate in a larger attack
	b.Side-channel attacks	The attacker exploits vulnerabilities in the physical hardware or software of the edge device or sensor to extract information or perform unauthorized actions
	c. DoS attacks	The attacker floods the edge device or sensor with requests or data to overwhelm its capacity, causing it to become unresponsive or shut down completely
Fog computing	a. Data interception attacks	The attacker intercepts data flowing between fog nodes or between fog nodes and edge devices, potentially accessing sensitive data or performing unauthorized actions
	b. Man-in-the-middle attacks	The attacker intercepts communication between two parties, such as a fog node and an edge device, to eavesdrop on or manipulate the communication
	c. Cryptographic attacks	The attacker exploits vulnerabilities in cryptographic protocols or implementations to access sensitive data or perform unauthorized actions
	d. Traffic analysis attacks	The attacker analyzes network traffic to deduce information about the communication patterns or content, potentially exposing sensitive information
Cloud computing	a. DDoS attacks	The attacker floods a cloud service with requests or data from multiple sources to overwhelm its capacity, causing it to become unresponsive or shut down completely
	b. Data breaches and theft	The attacker gains unauthorized access to cloud data and steals sensitive data, such as personal or financial information
	c. Insider attacks	The attacker is an authorized user with access to cloud resources and uses this access to perform unauthorized actions, such as stealing data or disrupting services
	d. VM hopping attacks	The attacker exploits vulnerabilities in VM technology to move from one VM to another within a cloud environment, potentially accessing sensitive data or performing unauthorized actions

M-BFS outperforms other frameworks, the results indicate that they all have some level of resilience against the concerned attack variants, which is a positive indication of their use in dynamic IoT infrastructures.Similarly, Fig. 4(b) provides some insights into the susceptibility of various frameworks to different types of attacks in the context of fog computing security in dynamic IoT infrastructures. It is noted that the M-BSF has the highest ARR (lowest susceptibility) to all the fog-level attack variants, which is97.47% for data interception attacks, 98.32% for MM attacks, 97.32% for cryptographic attacks, and 96.15% for traffic analysis attacks. Overall, the proposed framework has registered average differences of 4.70% susceptibility against all other popular frameworks. Likewise, Fig. 4(c) also depicts the performance of the proposed M-BSF regarding different potential attacks on the cloud-level security process. M-BSF has a high ARR for DDoS attack types (98.32%) while registering 95.45% for insider threats. But compared to other frameworks, the proposed security model excels over the existing models against all four attacks at the cloud-level security process. This indicates that the proposed M-BSF have a greater level of resilience against various types of attacks at each stage of the dynamic IoT environment, which exposes the positive indication of their use in any dynamic critical infrastructures.

4.2.2 DPI

The Data Preservation Index (DPI) is a quantitative measure of the degree to which digital data is being preserved over time. It provides an assessment of the quality and completeness of the preservation of data in a digital repository or archive. The DPI is measured based on various factors like fixity (F), completeness (Þ), usability (u), and integrity (I). Fixity is used to measures the extent to which the data has remained unchanged since it was first

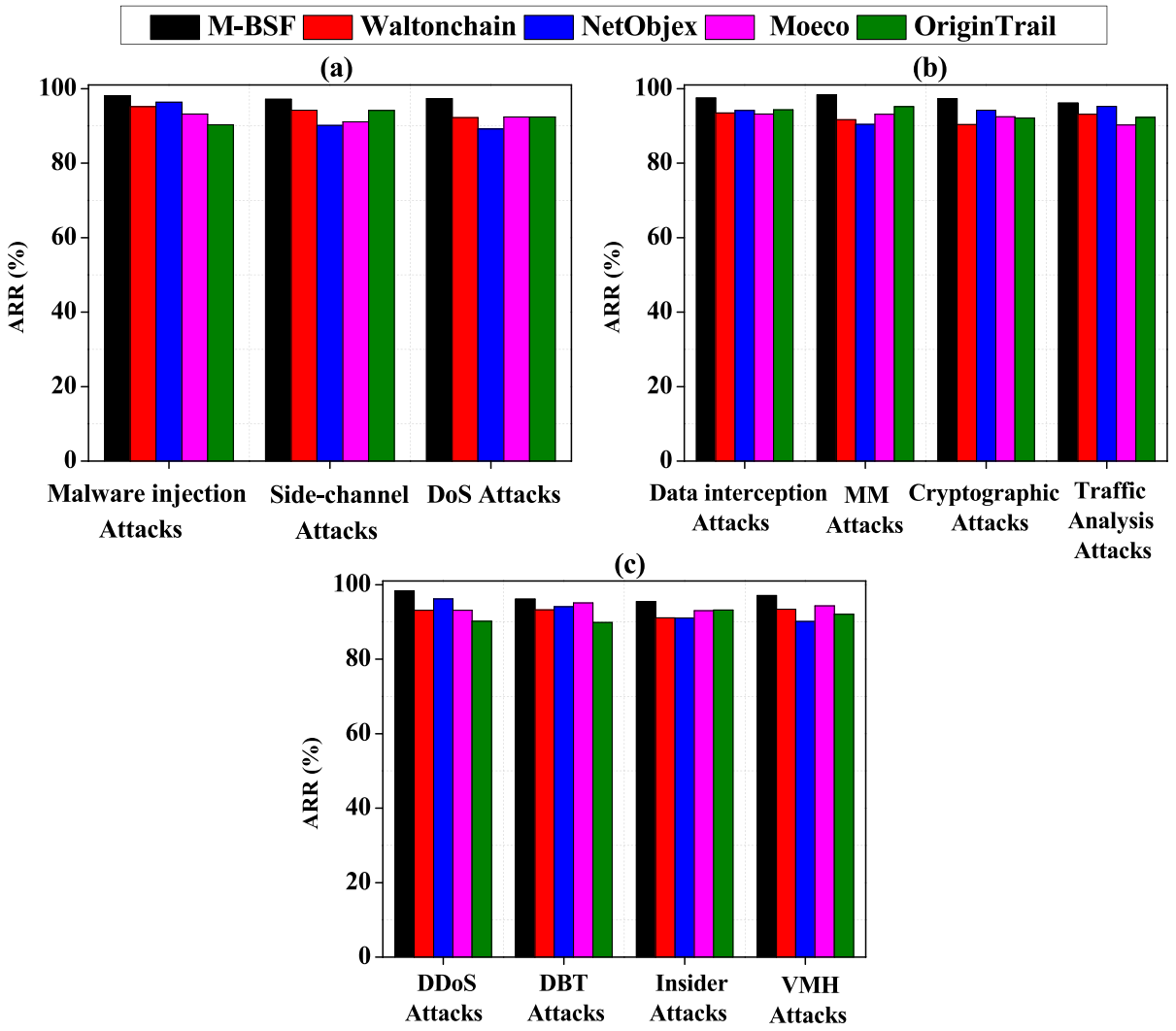

Fig. 4 Analysis of ARR against various attacks at (**a**) Edge Computing level, **b** Fog Computing Level, and **c** Cloud Computing Level

deposited in the repository. It involves verifying that the digital object has not been altered in any way. Completeness measures the degree to which the data set is complete, as compared to the original data set (kumar et al., 2011). It involves checking for any missing files, metadata, or other components of the data set. Usability measures the degree to which the data can be used by researchers, both now and in the future. It involves assessing the accessibility, format, and quality of the data. Integrity measures the degree to which the data has remained intact and accurate over time, t. It involves assessing the accuracy and completeness of the metadata associated with the data set. Thus based on all this factors, the DPI is equated as,

$$\boldsymbol{DPI} = \sum_{t=0}^{T} [\boldsymbol{F} + \mathrm{FX} + \mathbf{u} + \mathbf{I}] \tag{26}$$

The DPI statistics provided in the Fig. 5 are an indication of the security levels of different IoT frameworks. The DPI values can be classified as good, fair, or poor depending on their values. Based on the observed statistical data, the security level for each framework can be technically elaborated as follows:

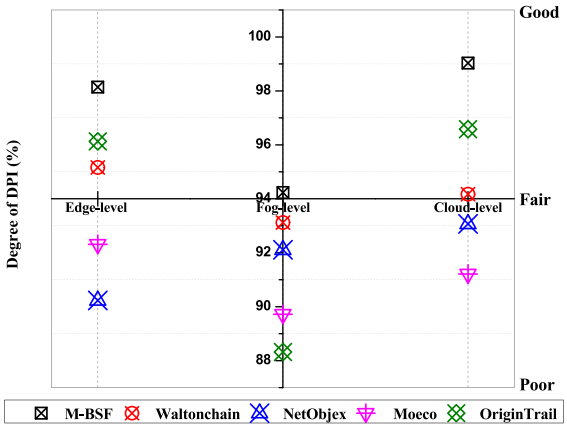

Fig. 5 Analyzing the Effectiveness of Data Preservation Practices

M-BSF: The DPI values for M-BSF are good for edge-level and cloud-level security, with a fair score for fog-level security.
Waltonchain: The DPI values for Waltonchain are fair to good for all three levels of security, with the highest score at the edge-level.
NetObjex: The DPI values for NetObjex are fair to poor for all three levels of security, with the highest score at the fog-level.
Moeco: The DPI values for Moeco are fair to good for all three levels of security, with the highest score at the edge-level.
OriginTrail: The DPI values for OriginTrail are good for all three levels of security, with the highest score at the edge-level and cloud-level.

In general, the frameworks with good security levels have higher DPI scores across all three levels, while those with fair or poor security levels have lower DPI scores.

4.2.3 Computation Overheads

Computation overhead in IoT security refers to the additional computational resources required to implement security mechanisms and protocols in an IoT system. The computation overhead in IoT security depends on factors such as the number of devices in the network, the complexity of the security mechanisms, the amount of data being transmitted, and the processing capabilities of the devices. Thus, the computation overhead (O) is defined as,

$$O = 10^{E} \sum_{i=1}^{N} \left(\tau_i \times \upsilon_i\right) \tag{27}$$

where, N is the total number of IoT devices in the network, τ_i is the computational cost (in terms of CPU cycles) required for device i to perform a security operation, such as encryption, decryption, or authentication, υ_i is the total number of security operations that device i needs to perform during a given time period, such as a second or a minute and E denotes the constant that represents the order of magnitude of the computation overhead, typically ranging from 6 to 9 depending on the size of the IoT network and the complexity of the security operations.

Figure 6(a), (b), (c), and (d) shows the computation overheads for five different frameworks [3, 21, 28] across four different transaction volumes (250, 500, 750, and 1000) in a dynamic IoT infrastructure. Computation overhead is the additional time and resources required to perform a particular computation or operation.

From the Table 7, we can see that the average computation overheads vary across different frameworks and transaction volumes. From the observation, it is inferred that the proposed M-BSF has the lowest average computation overheads across all transaction volumes. This suggests that M-BSF may be a more efficient framework in terms of computation resources. Waltonchain has the highest average computation overheads across all transaction volumes.

This suggests that Waltonchain may require more computation resources to perform the same operations as the other frameworks. Across all frameworks, we can see that the average computation overheads increase as the transaction volume increases. This is expected, as more transactions require more computation resources. There is some variation in the average computation overheads for different frameworks at the same transaction volume. For an instance, at a transaction volume of 750, M-BSF has an average overhead of 3.64, while OriginTrail has an average overhead of 5.95. This suggests that different frameworks may have different strengths and weaknesses in terms of computational efficiency.

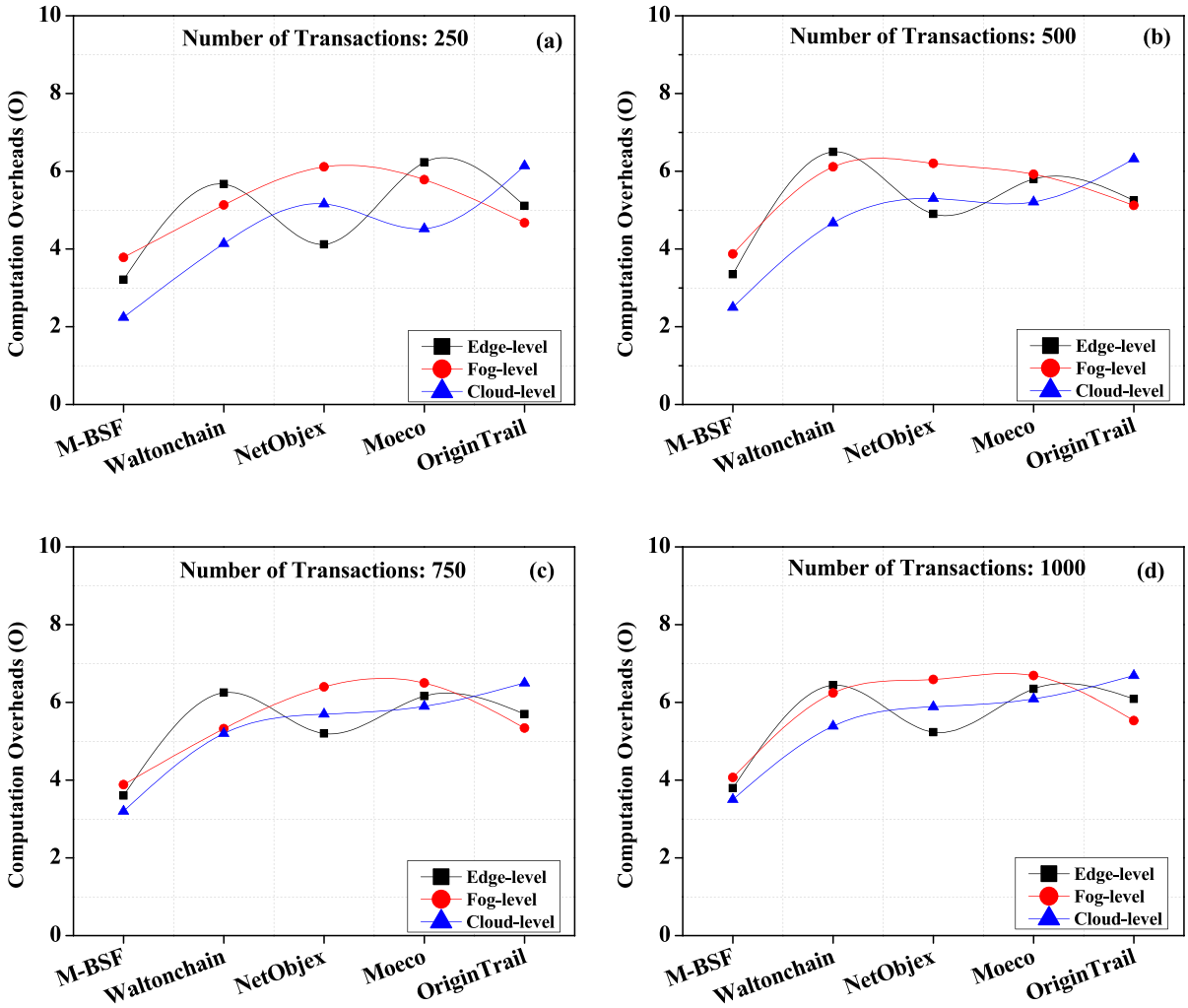

Fig. 6 Analysis of [O] for different Frameworks at three varying Security process of IoT Infra

Table 7 Average Computation Overheads

Frameworks	Transactions			
	250	500	750	1000
M-BSF	3.08	3.24	3.64	3.86
Waltonchain	4.98	5.35	5.59	6.02
NetObjex	5.13	5.30	5.59	5.96
Moeco	5.51	5.88	6.19	6.38
OriginTrail	5.31	5.56	5.95	6.20

4.2.4 Storage Overheads

In IoT security, storage overhead refers to the extra storage capacity required to store security-related data

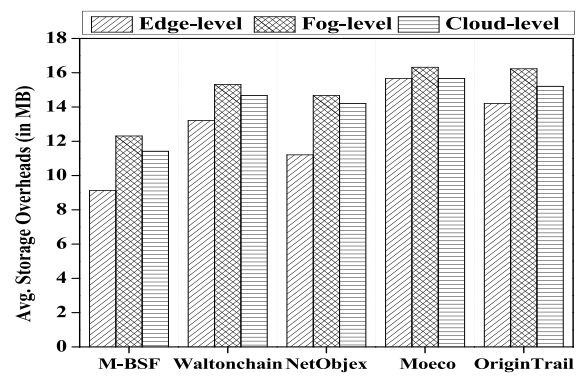

Fig. 7 Analysis of Average Storage Overheads

and metadata, such as encryption keys, access control policies, audit logs, and security event logs. Such storage overhead is optimized via cloud services. The mathematical computation for storage overhead in a blockchain-based IoT infrastructure can be estimated based on the significant factors like transaction data (TD), blockchain size (BS), and retention period (RP), which is expressed as,

$$S_o = (T_D + B_s) * R_p \tag{28}$$

The results from Fig. 7 represent the average storage overheads of five different IoT platforms across three levels of the IoT infrastructure: edge-level, fog-level, and cloud-level. From the resultants, it is noted that the storage overheads of each platform increase as we move from the edge-level to the fog-level to the cloud-level, which is expected since there is generally more storage available at higher levels of the infrastructure.

Looking at the platforms individually, we can see that M-BSF has the lowest average storage overhead at the edge-level, while Moeco has the highest. At the fog-level and cloud-level, Waltonchain has the highest average storage overhead among the platforms, while M-BSF has the lowest average storage overhead at the fog-level and OriginTrail has the lowest at the cloud-level. By using the Argon-2di hashing method in blockchain for cloud-level data storage, the data can be securely stored on the blockchain, and its integrity can be easily verified by anyone with access to the blockchain. The use of a unique salt value for each chunk of data makes it difficult for attackers to perform pre-computation attacks or rainbow table attacks. Additionally, the use of a secure and irreversible function to combine the hash values makes it difficult for attackers to reverse the hash and recover the original data.

5 Conclusion and Future Work

The research highlights the challenges of securing sensitive data in IoT environments and how blockchain technology can be leveraged to provide secure data sharing across disparate systems. Based on the evaluation of various metrics such as ARR (> 96%), DPI (with the highest degree), computational overhead (7 m), and storage overheads (13 MB), the M-BSF framework, which incorporates edge-level, fog-level, and cloud-level security via baby kyber and scaling kyber cryptosystems with robust hashing methods in blockchain, appears to be an acceptable alternative to current methodologies. It is also noted from the observations that the proposed framework yields higher trustability among users in addressing some common attacks prevalent at edge, fog, and cloud levels.Future enhancements include integrating decentralized identity systems to enhance user privacy and control over their data in IoT environments. Overall, the proposed M-BSF security architecture presents a promising approach to securing sensitive data in IoT environments, but further research and development are necessary to enhance its effectiveness and practicality.

Acknowledgements This research is supported by Princess Nourah bint Abdulrahman University, Researchers Supporting Project Number (PNURSP2023R151), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions All the authors contributed equally in the paper.

Data Availability Data is available upon request.

Declarations

This work is original and not have been published elsewhere in any form.

Competing interests The authors declare no competing interests.

References

1. Kaushal, R. K., Bhardwaj, R., Kumar, N., Aljohani, A. A., Gupta, S. K., Singh, P., Purohit, N.: Using mobile computing to provide a smart and secure internet of things (IoT) framework for medical applications. Wirel. Commun. Mob. Comput. 1–13 (2022). https://doi.org/10.1155/2022/8741357
2. Cheng, B., Zhu, D., Zhao, S., Chen, J.: Situation-aware IoT service coordination using the event-driven SOA paradigm. IEEE Trans. Netw. Serv. Manage. **13**(2), 349–361 (2016). https://doi.org/10.1109/TNSM.2016.2541171
3. Shetty, P., Kumar, M. R., Vyas, T., M, A., Gehlot, A., Pant, K.: Application of cryptographic methods to blockchain technology to increase data reliability. 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART) (2022). https://doi.org/10.1109/smart55829.2022.10047207
4. Lian, Z., Zeng, Q., Wang, W., Gadekallu, T. R., Su, C.: Blockchain-Based Two-Stage Federated Learning With Non-IID Data in IoMT System. IEEE Transactions on Computational Social Systems (2022). https://doi.org/10.1109/TCSS.2022.3216802
5. Alam, T., Gupta, R.: Federated learning and its role in the privacy preservation of IoT devices. Future Internet **14**(9), 246 (2022). https://doi.org/10.3390/fi14090246
6. BigQuery public datasets. (n.d.). Google Cloud. https://cloud.google.com/bigquery/public-data
7. Saab, S., Jr., Phoha, S., Zhu, M., Ray, A.: An adaptive polyak heavy-ball method. Mach. Learn. **111**(9), 3245–3277 (2022)
8. Saab, S., Jr., Fu, Y., Ray, A., Hauser, M.: A dynamically stabilized recurrent neural network. Neural Process. Lett. **54**(2), 1195–1209 (2022)
9. Kumar, L. R., Babu, A. S.: Genuine and forged offline signature verification using back propagation neural networks. Int. J. Comput. Sci. Inf. Technol. **2**(4). https://www.academia.edu/77622578/Genuine_and_Forged_Offline_Signature_Verification_Using_Back_Propagation_Neural_Networks
10. Kumar, P. M., Rawal, B., Gao, J.: Blockchain-enabled Privacy Preserving of IoT Data for Sustainable Smart Cities using Machine Learning. 2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS). https://doi.org/10.1109/comsnets53615.2022.9668530
11. Rakic, B., Levak, T., Drev, Z., Savic, S., Veljkovic, A.: First purpose built protocol for supply chains based on blockchain (2017). https://origintrail.io/storage/documents/OriginTrail-White-Paper.pdf
12. Karthick Raghunath, K. M., Koti, M. S., Sivakami, R., Vinoth Kumar, V., NagaJyothi, G., Muthukumaran, V.: Utilization of IoT-assisted computational strategies in wireless sensor networks for smart infrastructure management. Int. J. Syst. Assur. Eng. Manag. (2022) https://doi.org/10.1007/s13198-021-01585-y
13. Pustišek, M., Kos, A.: Approaches to Front-End IoT Application Development for the Ethereum Blockchain. Procedia Comput Sci **129**, 410–419 (2018). https://doi.org/10.1016/j.procs.2018.03.017
14. Rani, S., Babbar, H., Srivastava, G., Gadekallu, T. R., Dhiman, G.: Security Framework for Internet of Things based Software Defined Networks using Blockchain. IEEE Internet Things J. (2022)
15. Hong, W., Cai, Y., Yu, Z., & Yu, X.: An Agri-product traceability system based on IoT and blockchain technology. 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN) (2018). https://doi.org/10.1109/hoticn.2018.8605963
16. Lv, Z., Qiao, L., Hossain, M.S., Choi, B.J.: Analysis of using blockchain to protect the privacy of drone big data. IEEE Network **35**(1), 44–49 (2021). https://doi.org/10.1109/MNET.011.2000154
17. Shetty, P., Kumar, M. R., Vyas, T., M, A., Gehlot, A., Pant, K.: Application of cryptographic methods to blockchain technology to increase data reliability. 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART) (2022). https://doi.org/10.1109/smart55829.2022.10047207
18. Chen, P., Liu, H., Xin, R., Carval, T., Zhao, J., Xia, Y., ... Zhao, Z.: Effectively detecting operational anomalies in large-scale IoT data infrastructures by using a GAN-based predictive model. Comput. J. **65**(11), 2909-2925 (2022). https://doi.org/10.1093/comjnl/bxac085
19. Digital twin asset management with IoT, AI, Blockchain for Automation: Digital Twin Asset Management with IoT, AI, Blockchain for Automation (n.d.). https://www.netobjex.com/
20. Durga Indira N, Venu Gopala Rao M.: Red deer optimization for automatic modulation classification using hybrid extreme learning machine with bagging classifier. Int. J. Model Simul Sci Comput. **13**(06) (2022a). https://doi.org/10.1142/s1793962322500635
21. Lee, S., Bae, M., Kim, H.: Future of IoT networks: a survey. Applied Sciences **7**(10), 1072 (2017). https://doi.org/10.3390/app7101072
22. Rajasekaran, A.S., Azees, M., Al-Turjman, F.: A comprehensive survey on blockchain technology. Sustain Energy Technol Assess **52**, 102039 (2022). https://doi.org/10.1016/j.seta.2022.102039
23. Ozyilmaz, K.R., Yurdakul, A.: Designing a Blockchain-Based IoT with Ethereum, Swarm, and LoRa: The Software Solution to Create High Availability With Minimal Security Risks. IEEE Consum Electron Mag **8**(2), 28–34 (2019). https://doi.org/10.1109/mce.2018.2880806
24. Numenta Anomaly Benchmark (NAB). (n.d.). http://www.kaggle.com. Retrieved March 14, 2023, from https://www.kaggle.com/datasets/boltzmannbrain/nab
25. Qu, Z., Zhang, Z., Liu, B., Tiwari, P., Ning, X., ... Muhammad, K.: Quantum detectable Byzantine agreement for distributed data trust management in blockchain. Inf. Sci. **637**, 118909 (2023). https://doi.org/10.1016/j.ins.2023.03.134
26. Cebe M, Kaplan B, Akkaya K (2018) A Network Coding Based Information Spreading Approach for Permissioned Blockchain in IoT Settings. Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. https://doi.org/10.1145/3286978.3286984
27. Durga Indira, N., VenuGopalaRao, M.: Deep learning CNN-based hybrid extreme learning machine with

bagging classifier for automatic modulation classification. Int. J. Intell. Syst. Appl. Eng. **10**(2s), 134–141 (2022). (**ISSN: 2147-6799**)
28. Moeco documentation | Moeco Documentation. (n.d.). Docs.moeco.io. Retrieved March 24, 2023, from https://docs.moeco.io/
29. Cao, B., Wang, X., Zhang, W., Song, H., Lv, Z.: A many-objective optimization model of industrial internet of things based on private blockchain. IEEE Network **34**(5), 78–83 (2020). https://doi.org/10.1109/MNET.011.1900536
30. Judith, A. M., Priya, S. B., Mahendran, R. K., Gadekallu, T. R., Ambati, L. S. Two-phase classification: ANN and A-SVM classifiers on motor imagery BCI. Asian J. Control. (2022). https://www.researchgate.net/publication/366400731_Two-phase_classification_ANN_and_A-SVM_classifiers_on_motor_imagery_BCI
31. Waltonchain, EN.: Waltonchain Whitepaper 2.0 Official Release Announcement. Medium (2018). https://waltonchain-en.medium.com/waltonchain-whitepaper-2-0-official-release-announcement-1c819aeb0314
32. Wang, S., Sheng, H., Zhang, Y., Yang, D., Shen, J., ... Chen, R. Blockchain-empowered distributed multi-camera multi-target tracking in edge computing. IEEE Trans. Industr. Inf. (2023). https://doi.org/10.1109/TII.2023.3261890
33. Yaïci, W., Krishnamurthy, K., Entchev, E., Longo, M.: Recent advances in Internet of Things (IoT) infrastructures for building energy systems: a review. Sensors **21**(6), 2152 (2021). https://doi.org/10.3390/s21062152
34. Yohan, A., Lo, N.-W. An over-the-blockchain firmware update framework for IoT devices. 2018 IEEE Conference on Dependable and Secure Computing (DSC) (2018). https://doi.org/10.1109/desec.2018.8625164.