future internet

MDPI

Review

Internet of Things and Blockchain Integration: Security, Privacy, Technical, and Design Challenges

Yehia Ibrahim Alzoubi [1], Ahmad Al-Ahmad [1,*], Hasan Kahtan [2] and Ashraf Jaradat [1]

[1] Management Information Systems Department, College of Business, American University of the Middle East, Egaila 15453, Kuwait; yehia.alzoubi@aum.edu.kw (Y.I.A.); ashraf.jaradat@aum.edu.kw (A.J.)
[2] Cardiff School of Technologies, Cardiff Metropolitan University, Cardiff CF5 2YB, UK; hkahtan@cardiffmet.ac.uk
* Correspondence: ahmad.alahmad@aum.edu.kw

Abstract: The Internet of things model enables a world in which all of our everyday devices can be integrated and communicate with each other and their surroundings to gather and share data and simplify task implementation. Such an Internet of things environment would require seamless authentication, data protection, stability, attack resistance, ease of deployment, and self-maintenance, among other things. Blockchain, a technology that was born with the cryptocurrency Bitcoin, may fulfill Internet of things requirements. However, due to the characteristics of both Internet of things devices and Blockchain technology, integrating Blockchain and the Internet of things can cause several challenges. Despite a large number of papers that have been published in the field of Blockchain and the Internet of things, the problems of this combination remain unclear and scattered. Accordingly, this paper aims to provide a comprehensive survey of the challenges related to Blockchain–Internet of things integration by evaluating the related peer-reviewed literature. The paper also discusses some of the recommendations for reducing the effects of these challenges. Moreover, the paper discusses some of the unsolved concerns that must be addressed before the next generation of integrated Blockchain–Internet of things applications can be deployed. Lastly, future trends in the context of Blockchain–Internet of things integration are discussed.

Keywords: BC; BIoT; challenge; integration; trend

Citation: Alzoubi, Y.I.; Al-Ahmad, A.; Kahtan, H.; Jaradat, A. Internet of Things and Blockchain Integration: Security, Privacy, Technical, and Design Challenges. *Future Internet* **2022**, *14*, 216. https://doi.org/10.3390/fi14070216

Academic Editor: Claude Chaudet

Received: 30 June 2022
Accepted: 20 July 2022
Published: 21 July 2022

1. Introduction

Academics, researchers, and entrepreneurs are all interested in the Internet of things (IoT) these days because of its ability to provide novel services across a wide range of applications such as COVID-19 [1] and intelligent healthcare [2]. The IoT connects diverse things and devices to establish a physical network where processing, sensing, and communication activities are automatically managed without human intervention [3]. In the last decade, there has been significant growth in the number of IoT devices on the market. The number of IoT devices on the market is nearing 25 billion, with predictions that this number will rise to 50 billion by the end of 2025 [4]. To achieve such massive development, new arrangements are required, such as following the centralized IoT–cloud approach. The organization's system for cloud computing and data processing is referred to as IoT–cloud architecture. Here, the cloud manages and visualizes data flows from IoT devices while processing and analyzing them [5]. Although this approach may work well today, the projected development indicates that new approaches will be required in the future [6,7] due to several challenges of the centralized IoT–cloud approach. The following are some of the problems that a centralized IoT–cloud architecture faces [8–12]: (1) if the centralized server fails, the entire network system is at risk of being paralyzed and interrupted; (2) data fraud makes it difficult for IoT devices owners to trust partners who have oversight and access to the collected data; (3) data maintained in centralized clouds lack accountability and traceability because they depend on a trusted third party to store and retain data;

(4) the central architecture is no longer robust enough to handle vast amounts of data and end-to-end interactions as a result of the rapid growth of IoT applications. Furthermore, due to the variety of smart devices on the market, maintaining and updating these devices are almost impossible; (5) since transparency is critical for promoting security and trust when designing next-generation IoT solutions, open-source techniques should be considered; (6) because most safe cryptographic methods need a large amount of processing power and energy, the encryption process is a big barrier due to the heterogeneity and limited compute capacity of IoT devices; (7) due to the constantly increasing number of IoT devices number, relevant IoT ecosystems must accommodate future network development while also processing a huge volume of data exchanged in a high-performance manner. The abovementioned challenges cannot be achieved in centralized IoT–cloud architecture. These problems necessitate a rethinking of the IoT's structure. Although decentralized systems for enormous peer-to-peer (P2P) wireless sensor networks have been developed in the past to overcome the drawbacks of the centralized IoT–cloud architecture, security and privacy requirements were lacking until the advent of Blockchain (BC) technology [13].

BC can carry out, organize, and monitor transactions provided by several devices without the need for a centralized cloud. To validate a transaction, BC is a decentralized system that does not require trust among participants. Its origins can be traced back to the cryptocurrency Bitcoin system [14]. The integration of BC and IoT (BIoT) may result in several benefits [15–18] for IoT security. Firstly, it provides a P2P framework that does not require a middle layer such as a third trusted party. Secondly, BC technology has no single point of failure, and, when it is used with smart contracts, it enables more secure transactions, which protects against various scams since smart contracts provide access control and improve stability, confidentiality, and authentication. By ensuring the data are cryptographically encrypted and signed by the rightful sender, BC ensures data confidentiality and authentication. Thirdly, the capacity of the entire network can be expanded due to its P2P nature. Fourthly, BC enables transactions to be performed quickly. Once built and attached to the BC network, each IoT device will get symmetric key pair, eliminating the need for key management and delivery in the BC network. As a consequence, lightweight authentication protocols can be used. The need for computing and memory capacity in IoT devices can be met and organized by these lightweight protocols. Fifthly, the immutability of IoT using data logs stored on BC ensures traceability and transparency. Lastly, due to its tamper-proof design and safe storage, BC may enable the secure release of software updates to IoT devices.

Despite all of the above advantages, BIoT integration may pose several challenges [3,19–23]. Mission-critical situations, in particular, raise additional questions. Since BIoT integration is a dynamic process influenced by multiple interrelated factors, adding BC to the IoT ecosystem adds more organizational and technological requirements. Hence, this paper aims to provide a comprehensive survey about BIoT integration challenges. Consequently, this paper focuses on the following research questions:

RQ1: What are the current challenges that face BIoT integration?
RQ2: What recommendations have been provided in the literature to overcome these challenges?
RQ3: What are the future concerns and research trends for BIoT integration?

This paper has several key contributions. Firstly, although BC has been in use for some years, little thorough research on BIoT challenges exists. This article provides a comprehensive survey of BIoT challenges based on a survey of the existing literature on BIoT integration. This survey might be one of several studies that look into these issues in depth. Seven challenges categories were identified in this paper: security, privacy preservation, technical considerations, scalability, computational processing, characteristic considerations, regulations and guidelines, and BIoT design. Secondly, in addition to exploring the problems of BIoT, this paper also reports the recommended solutions to these challenges. The majority of reviewed studies recommended turning the BC phase into another layer (i.e., not an IoT device), such as a fog layer, edge layer, software-defined networking (SDN),

or cloud layer as IoT is still in its infancy. Thirdy, the current research problems of IoT decentralization using BC, as well as future challenges in the field, are presented in depth. Seven future BIoT challenges were identified: security, privacy, communication and consensus mechanism, scalability and capability constraints, standardization and regulations, BC platform, and big data. Lastly, potential BIoT integration trends were explored. New BC-based business models, AI, quantum computing, double-chained IoT security, and IoT, BC, and 6G interoperability are among these trends.

The paper performed a thorough study of the literature by searching the related papers in major scholarly databases (IEEE Xplore, Springer Link, Wiley Online Library, ACM digital library, MDPI Online, Science Direct, Emerald Insight, and SAGE Publication) to identify and consider the current state of BIoT challenges, as well as future research directions. According to the databases scanned, the use of various BC platforms has been the subject of several journal and conference publications in the field of BIoT such as Bitcoin, Ethereum, Multichain, and Hyperledger Fabric. Table 1 summarizes the abbreviations that appeared in this article. The remainder of the article is structured as follows: Section 2 presents the background and related work; Section 3 responds to the research questions; future research trends and directions are explored in Section 4; the findings' discussion and limitations are presented in Section 5; Section 6 concludes the paper.

Table 1. Table of abbreviations used in the article.

Abbreviation	Definition	Abbreviation	Definition
ADEPT	Autonomous Distributed P2P Telemetry	FNN	Feed-forward neural network
AES	Advanced Encryption Standard	HLF	Hyperledger Fabric
BBP	Bitcoin Backbone Protocol	IATBA	International Association for Trusted Blockchain Applications
BC	Blockchain	IoT	Internet of things
BECI	Bitcoin Energy Usage Index	IOTA	Internet of things application
BIoT	Blockchain and Internet of things integration	IPFS	InterPlanetary File System
CBECI	Cambridge Bitcoin Power Consumption Index	ISO	International Organization for Standardization
CNN	Convolutional neural network	LSTM	Long-short term memory
DAG	Directed acyclic graph	MiTM	Man-in-the-middle
DDoS	Distributed denial of service	NIS	Network and information security
DID	Decentralized identity	PBFT	Practical Byzantine fault tolerance
DHT	Distributed hash-table	PKI	Public key infrastructure
DLT	Distributed ledger technology	PoA	Proof-of-activity
DPoS	Delegate proof-of-stake	PoC	Proof-of-capacity
EEA	Enterprise Ethereum Alliance	PoW	Proof-of-work
ECC	Elliptic Curve Cryptography	QoS	Quality of service
ETH	Ether	SDN	Software-defined networking
EVM	Ethereum Virtual Machine	SSI	Self-sovereign identity
FBA	Federated Byzantine agreement	SVM	Support vector machines
FC	Fog computing	zkSNARK	Zero-knowledge succinct non-interactive argument of knowledge

2. Background and Related Work

BC's novelty is one of the reasons that more professionals and academics are becoming interested in it. The structure and architecture of the BC are discussed and illustrated in the sections below. The basic features of BC, as well as three different types of BC (private, public, and consortium), are then explained in detail. It is important to note that the structure and architecture discussion is centered on Bitcoin, which is the most prevalent and documented BC in the literature. We also cover Ethereum, Hyperledger Fabric (HLF), and Multichain BC-based platforms.

2.1. Blockchain Structure and Architecture

Distributed ledger technology (DLT) and directed acyclic graph (DAG) are revolutionizing the way information is shared [24]. DLT is a P2P network that maintains a decentralized database [25]. The ledger is validated and copied by each node. The BC is one kind of DLT. The BC divides data into blocks, which are subsequently chained together (connected) using an append-only structure. Although it is far from the only DLT data format, the chain-based block structure is the most widespread [26]. DLT may also be implemented using other data structures such as DAG. DAG, like BC, may store transactions. Nodes connected to at least one, but possibly many additional transactions describe these data transactions. Links, on the other hand, are precisely directed, pointing from a previous transaction to a current one. It is also worth noting that because DAGs are acyclic, they do not allow loops [27]. There are no blocks in DAGs, and no mining is conducted, compared to BC. While transactions may authenticate one another, they cannot validate themselves. Furthermore, while entering the DAG, at least one prior transaction must be authenticated before a new transaction may be created. Each new transaction must refer to the previous one [27]. The hashes of the parent transaction are signed by the new transaction, which then integrates them into the new transaction [28]. DAG and BC technologies are combined in hybrid DLTs. Bexam is a hybrid DLT that combines flexible chains with hierarchical nodes to give the security of BC, generating roughly 40 million transactions per second. Bexam is highly scalable and simple to incorporate into large-scale systems. Furthermore, processing resources and power consumption are low. Token technology is also used by Bexam to create transactions [28]. We focus on BC in this paper because it is the most widely used distributed ledger system.

BC is a distributed and mutual ledger that keeps track of a constantly expanding list of blocks that are connected and guarded with cryptography. It is necessary to understand BC elements before diving into the activities of BC. The BC employs a decentralized architecture, with the user and data access permissions separated. The security problems associated with central controls are eliminated with BC-based applications [29]. To maintain data privacy and security, all processes are registered [30]. The BC, in a typical Bitcoin structure, is made up of three technical elements: a cryptographic hash function, a Merkle tree, and a BC. Hash functions are mathematical formulas that produce a lengthy sequence of characters as inputs. All of the previous inputs are combined into a single Merkle Tree, which links all of the transactions into the BC [31]. The block header is made up of the Merkle tree, the block stamp, and the preceding block header, which is a special ID. As a result, the BC uses the block header to track previous record history. The Merkle tree is a data structure for storing key-value pairs that are encrypted.

The Bitcoin BC system creates a secure public data reading process dependent on anonymity. Both nodes can instantly and safely validate and share data inside the device without any interference, thanks to agreements and protocols. The data in the BC system are encrypted and anonymized to different degrees. The most widely used hash functions in BC and cryptocurrencies are the SHA-256 series. There is no requirement to reveal or check each node's identity or relevant information unless lawfully necessary. Before being written into the BC, data must be checked with a timestamp (used to assure the uniqueness of the transactions). All can write and read data and nodes with maintenance functions in

the BC system, which is free and transparent. Via open interfaces, anybody can query BC data and create similar applications.

When any participant or node in the network makes a transaction, it is broadcasted to all nodes in the network after a so-called signing process, which involves two steps: hashing and encryption to generate a digital signature. The process starts by hashing the transaction with some hash function, such as SHA-256, which yields the hash value. Then, the sender's private key encrypts the transaction, resulting in a digital signature. After that, the transaction, as well as the digital signature, are transmitted to the entire network [32,33]. A BC validator/miner is in charge of checking transactions on the network. After collecting a series of transactions, miners can begin the validation process by performing the following steps to ensure they are legal (i.e., no malicious transactions or double spends) [21]: (1) miners decode the digital signature using the sender's public key, resulting in a decrypted hash value; (2) the miners use the same hash function to generate a new hash value from the received address; (3) if the current hash value meets the decrypted hash value, the transaction has not been tampered with, and the integrity, verification, and non-repudiation requirements have been met; (4) each miner creates a block of authenticated transactions, and that the validation process is accordingly finished [21,34].

The hardware layer, data layer, network layer, consensus layer, incentive layer, contract layer, and application layer are the seven major layers in a basic BC architecture [26,27,35,36]. The elements of each layer are depicted in Figure 1. In this section, we go through each of these levels and their functions.

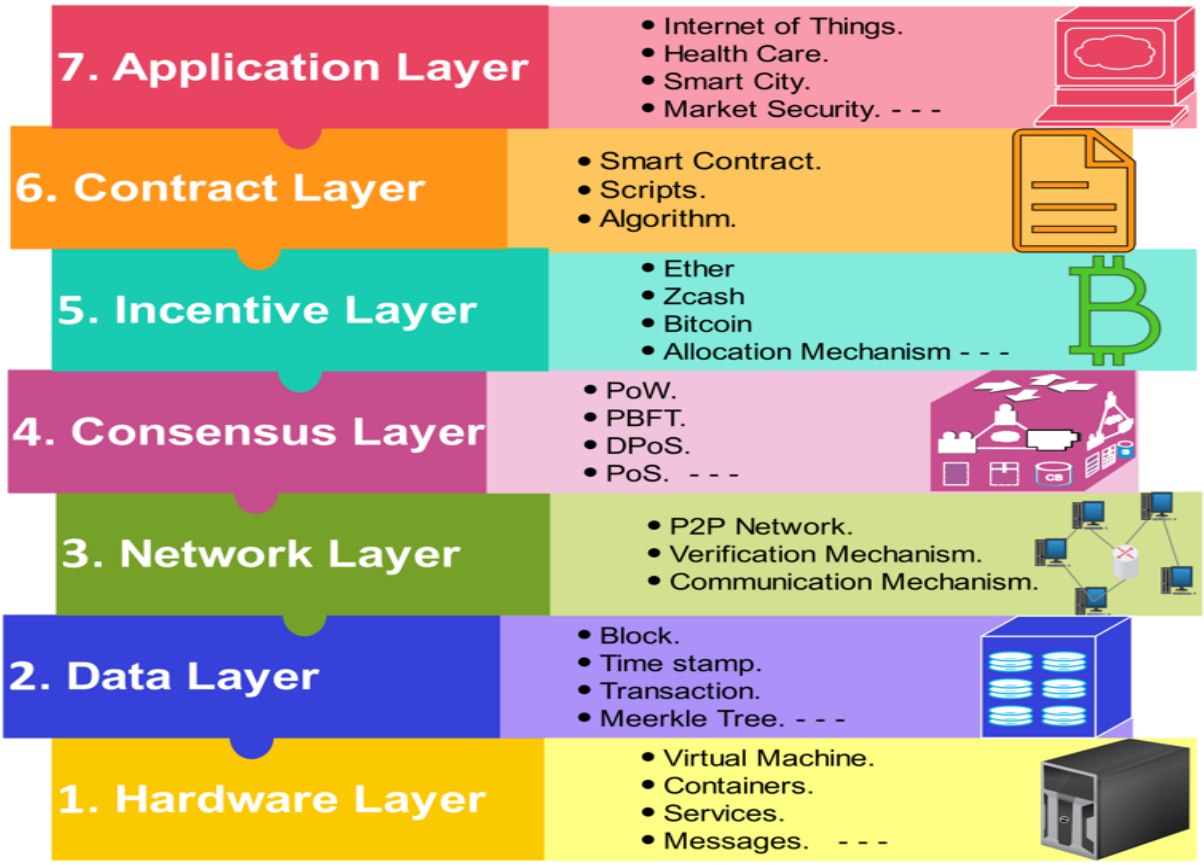

Figure 1. BC layered architecture.

(1) Hardware layer: Actuators, sensors, smart devices, controllers, and edge/fog nodes are all represented in this layer. The IoT is made up of these devices connected by a variety of wireless and wired communication protocols.

(2) Data layer: Blocks, transactions, the hash function, the digital signature, and the Merkle tree are all part of this layer. This layer collects IoT data from the lower layer

in the form of transactions and encrypts it using asymmetric cryptographic methods, hashes, and digital signatures.

(3) Network layer: This layer serves as a P2P network on top of the communication layer. Only a network architecture that allows peers to trade resources without the participation of a third party allows for decentralization. While all P2P participants can operate as both a requestor and service providers, they can be divided into categories on the basis of the support services they provide, such as database, routing, and mining.

(4) Consensus layer: The distributed consensus necessary to verify a block's trustworthiness and guarantees that all peers have an accurate ledger copy are managed by this layer. However, owing to network failures, communication delays, or malevolent nodes, agents and nodes may end up with various perceptions of the system's status (i.e., forks). As a result, avoiding such forks is one of the problems of a consensus method.

(5) Incentive layer: The incentive layer is the heart of the BC network since it includes economic factors such as Ether (ETH) (a cryptocurrency that was created as a result of the confirmation of transactions on the Ethereum), Zcash (a protocol that provides a decentralized cryptocurrency, to store funds and generate a new private key for every new account [21]), and allocation methods to incentivize nodes to give their time and effort to data verification.

(6) Contract layer: This layer is in charge of digital money, as well as the design and management of smart contracts. Algorithms, smart contracts, and scripts are applied to allow more sophisticated transactions.

(7) Application layer: This layer offers services across a wide range of industries, including logistics, healthcare, IoT, and smart cities.

2.2. Blockchain Platforms

In general, two major categories can be identified for BC: permissionless and permissioned [25–27].

(1) Permissionless: This form of BC, also known as public BC, permits transactions to be viewable to all nodes. To authenticate a transaction, every node in the network can participate in BC consensus. The node does not require authorization, and it may be unknown to the rest of the network. Nodes in a permissionless network support and collaborate on a large scale. Each transaction is associated with a processing fee, which offers an incentive for peers looking to add additional blocks to the BC [37]. Because altering the contents of the permissionless BC would be prohibitively costly, it is immune to hacking. Each transaction comprises an incentive (i.e., transaction fees) to the peer that approves the transaction into a new block because the decentralized consensus involves hundreds of other peers [36]. Bitcoin is the most well-known permissionless cryptocurrency. Another well-known permissionless BC is Ethereum.

(2) Permissioned: This type of BC network may be classified as either private or consortium BCs.

   A. Private: These BCs are generally located in the heart of a single company that can verify transactions. Transactions may be read by the public or authorized parties. Private BCs operate without the need for money or tokens, and their transactions are fee-free [38]. Because blocks are broadcasted by surrogate nodes, a private BC is not as impenetrable to tampering as a public BC, but the firm may roll back its BC at any point in time. Multichain is an example of a private BC. Multichain is a Bitcoin fork with several features, including rights management, rapid setup, and data streams [39].

   B. Consortium: This type of BCs is managed by a small cluster of users from outside the group who are not allowed to confirm transactions. While the whole public may view transactions, only members of a limited group can write them. HLF is the most widely used and well-known federated BC. There are two sorts of HLF nodes: validating peers and nonvalidating peers. Validating

peers are in charge of verifying transactions, establishing agreements, and keeping the ledger up to date. Nonvalidating peers can examine and verify transactions [36,40].

Due to the benefits that this technology provides, BC systems and implementation have recently arisen from a wide range of fields such as IoT, transportation, finance, eHealth, and energy applications. In the sections below, we survey some of the most widely used BC platforms. Since there are many platforms and they are constantly changing, it is difficult to study them all; thus, only the most common and most appropriate IoT domain platforms are surveyed (i.e., Bitcoin, Ethereum, HLF, and Multichain) [41]. Because investing in a specific BC technology is a mid- to long-term commitment, these are quite high levels of support. Some current systems are supported by a large number of individual developers, while others are supported by companies. The Ethereum Foundation, for example, is a nonprofit organization established in Switzerland, while the Bitcoin project has a large open-source development community [41]. IBM and the Linux Foundation support HLF. Table 2 offers a summary of the most critical characteristics of these four platforms. It is worth noting here that different authors refer differently to the specifications of the four platforms. The summary in the table is based on what was reported in [10,13,42–44].

Table 2. Comparison of Bitcoin, Ethereum, HLF, and Multichain BCs' platforms.

Features	Bitcoin	Ethereum	HLF	Multichain
Access	Public	Public	Consortium	Private
Open source	Yes	Yes	Yes	Yes
Consensus	PoW	PoS/PoW	PBTF/SIEVE	PBTF/Ripple
Crypto currency	Bitcoin	ETH	None	Multicurrency
Smart contracts	Bitcoin script	Smart contract	Chain code	Smart filters
Special hardware requirement	No	No	No	No
Average transaction per second	7	15–20	3500	200–1000
Hashing algorithm	SHA-256	Ethash, KECCAK-256	SHAKE256, SHA3	SHA-256
ID management	No	No	Yes	Yes
Key management	No	No	Yes	Yes
Trustless operation	Yes	Yes	Trusted validators	Trusted validators
Data confidentiality	No	No	Yes	Yes
Authentication	No	Medium	High	High

Three of the BC platforms reported in Table 2 endorse cryptocurrency, which may become significant according to how reward schemes are configured for various applications. Since many C platforms are built on Bitcoin BC, they share features such as using the proof-of-work (PoW) consensus protocol, not having specific hardware to generate new blocks, and being written in C++. All platforms use smart contracts, except for Bitcoin which does not use smart contracts. All the platforms do not use special hardware preparations, while only HLF and Multichain provide ID and key management, enable data confidentiality, and require trusted validators to validate the transactions.

It is essential to discuss the concept and functions of the smart contract in the context of BC platforms because they are a requirement of many BIoT platforms [45]. A smart contract is defined as a computerized protocol that implements a contract's terms [46]. A smart contract's ability to implement or self-execute contractual clauses is one of its most important characteristics [47]. Furthermore, smart contracts have greatly added to the energy of BC, and this integration has resulted in the second generation of BCs (i.e., BC 2.0). In a trustworthy environment, a mix of automatically executed contracts and no centralized oversight has the potential to revolutionize the way business is conducted today [47]. In

essence, the smart contract code is stored on the BC, and each contract is known by a unique address, which users may access by sending a transaction to [28]. The BC consensus protocol ensures that the contract is executed correctly. Smart contracts provide several benefits, including cost savings, speed, accuracy, performance, and openness, which have prompted the development of a slew of new applications in a diversity of fields [48]. While Bitcoin includes a simple scripting language, it has proven inadequate, prompting the development of modern BC systems that provide Smart contract features [49].

Ethereum, the most common smart contract BC network, is a BC with a Turing-complete programming language that enables smart contracts and dispersed applications to be defined. Ethereum contracts are written in a stack-based bytecode language at a basic level called "Ethereum Virtual Machine (EVM) code" [50]. Financial smart contracts also require details about current events and states in the real world. The so-called oracles have this information. These institutions are essential for the efficient incorporation of smart contracts into the real world, but they add to the challenge by requiring authentication, confidentiality, and oracle trust [51].

Table 3 reports the advantages and disadvantages of these four platforms [10,13,42–44,46]. The fact that all four most common BC platforms are open-source is a major factor in their success. While HLF and Multichain can provide high scalability and authentication, Bitcoin can provide low levels of authentication and scalability, and Ethereum can provide medium levels of scalability and authentication. Both Bitcoin and Ethereum are linked to a 51% attack level, while this value is 33% for HLF and Multichain. The security level provided by Bitcoin and Ethereum is low since the data are accessible to the public; however, the level is medium for HLF and high for Multichain. The privacy level is high when using HLF and Multichain, but low for Ethereum and Bitcoin since there is no authentication applied and they are publicly accessible.

Table 3. Advantages and disadvantages of Bitcoin, Ethereum, HLF, and Multichain BCs' platforms.

Platform	Advantages	Disadvantages
Bitcoin	• Open-source • P2P network, which allows rapid worldwide payments • It is the most secure and widely used BC-based payment method	• Low scalability, low security, low privacy, low transaction rate • 51% attack • Loops and recursion are not supported by the Bitcoin script. • Limited decentralization
Ethereum	• Open-source • Decentralized P2P network	• Medium scalability, low security, low privacy, low transaction rate • 51% attack
HLF	• High privacy, high scalability, high transaction rate • Open-source cross-industry standardization • Sophisticated data queries and key-value pair data are supported	• Medium security • 1/3 fault node attack • Complex architecture and lack of use cases
Multichain	• Open-source, high security, high privacy, high scalability • Low latency and high throughput • High query functionality and decentralization	• Medium transaction rate • 1/3 fault node attack • No built-in support for smart contracts

2.2.1. Bitcoin Platform

The first cryptocurrency, as well as the first BC platform, was Bitcoin. It provides a safe payment system that can be embedded into applications to conduct financial transactions quickly and reliably. Autonomous machines can use Bitcoin to make micropayments and act as wallets in the BIoT [51]. Due to the transaction costs charged, it is not financially advantageous to process small payments (i.e., less than $) using the current banking and credit systems. However, with Bitcoin, that is now ridiculously simple. However, due to a variety of factors, including scalability, fees, and block size, the BC platform design

is not practical for facilitating micropayments [52]. BC solutions should allow many micropayments to be made without the transactions being logged on the main BC [53]. Therefore, the new BC approach recommends grouping up multiple minor transactions into one larger transaction [54].

When using BC for micropayments, applications are usually tied to the currency, which can be a disadvantage since inflation of the coin can have a detrimental impact on the application. When carrying out transactions, Bitcoin provides a scripting language that enables complex conditions to be set. However, as compared to other smart contract systems, the scripting is very limited [46]. To validate transactions, Bitcoin uses the PoW consensus protocol, which requires substantial resources and energy.

2.2.2. Ethereum Platform

Ethereum is a BC smart contract platform because it allows users to execute user-defined applications, decreases the number of tree structures, and increases scalability, stability, and usability [50]. It is a network that allows decentralized apps to be built on top of the BC. For various items, Ethereum utilizes diverse trees including a receipt tree, a status tree, and a transaction tree. Each of these trees has its responsibilities and activities to complete. The transaction tree is where information about transactions is stored, such as transaction occurrences, blocked transactions, and transaction requests. The receiving tree is in charge of keeping track of all activities. The account validity, account balance, and other facts are handled by the status tree. Ethereum also has a strong upper-level service/app creation interface in the context of the EVM language. The EVM acts as a sandbox, allowing for a separate execution environment. Ethereum was created with the Solidity programming language in mind [51]. The addition of smart contracts to the BC pushes it away from currencies and makes it easier to incorporate this technology into new fields. Ethereum is the most common platform for designing apps and involves a large community. This can be used to enforce a variety of smart contracts, including supply chain, banking, and shopping [10]. When using PoW, Ethereum needs substantial resources and energy. The majority of IoT implementations use Ethereum or are Ethereum-compliant. The simplest solution is to create a smart contract in which devices broadcast their policies and measurements that adapt to changes [46].

2.2.3. Hyperledger Fabric Platform

Hyperledger started as an IBM-led Open BC project [55]. It was later donated to the Linux Foundation's Hyperledger Project, which evolved into the new Hyperledger. Different regions need different networks and different forms of BC, according to Hyperledger's architecture goals [55]. HLF is an open-source platform that meets the general network characteristics, including asset association and trading party identity verification, sensitive transactions using detached transactions and identities, and confidential contract encryption. HLF provides an infrastructure that includes a range of modules, such as various smart contract engines because the use of BC must respond to different needs [43]. HLF, in particular, contains distinct BC frameworks and tools. Different consensus protocols are supported by each framework. PoW is not supported by HLF, and extra privacy security has yet to be included. It is also worth noting that HLF has multinational partnerships with a variety of companies.

Various projects linked to BC have been built on the Hyperledger network, including HLF. HLF develops enterprise BC applications (i.e., rules that can be customized to aid various consensus protocols) such as distributed ledger systems, smart contract engines, database libraries, graphical interfaces, and functionality libraries. In the BC-HLF, general-purpose languages can be used to build distributed applications [43]. The IBM Watson IoT Platform, which offers device management and allows data filtering and analysis, can provide data to the BC from IoT devices, for instance.

2.2.4. Multichain Platform

Private BC can be created and deployed using the Multichain framework. Multichain is a fork of Bitcoin BC that focuses on adding features like user consent management and improving data ledger functions [56]. Multichain also advocates both PoW and "Mining Diversity", a consensus protocol built on round-robin [43]. Mining Diversity's fundamental premise is that users in a private BC network are already "trusted" to a degree because they are recognizable individuals. Mining Diversity will, thus, provide reliable consensus and mining on the private BC network without relying on the computationally expensive PoW algorithm. However, unlike some other platforms, Multichain does not provide users with extra privacy [55]. Multichain makes use of an API that adds additional features to the heart of the Bitcoin API, allowing for the managing of portfolios, funds, permissions, and transfers, among other things. It also includes a command-line tool for interacting with the network, as well as a variety of clients that can communicate with the network using JSON-RPC, including Java, Ruby, Node.js, and C#. Multichain is a Bitcoin core fork, and its source code is 64-bit compatible [55].

2.2.5. Other Popular Blockchain Platforms

There are several other well-known BC platforms including IoT Application (IOTA), Libra, EOS.IO, IoT Chain, IoTeX, HDAC, Atonomi, and Hydrachain [10,57,58].

- IOTA: IOTA is a block-less distributed ledger based on the DAG. The tangle that records the transactions in the IOTA DAG is known as the tangle. IOTA can execute a greater number of microtransactions per second without incurring a charge. A new transaction in IOTA must use a PoW method to validate and approve two prior transactions. As the number of users in the IOTA network rises, the tangle becomes more efficient, quicker, more reliable, and secure.
- Libra: It is a decentralized BC that enables cryptocurrency and has a consistent value supported by low-volatility reserves such as fiat money. This platform was created to assist unbanked people with quick, safe, and scalable financial services. Libra's unified auditing services for controllers and validators represent one of its most important features. Libra includes a native programming language called Move that allows the creation of customized transactions and smart contracts that are secure, flexible, and verifiable.
- EOS.IO: It is a new BC protocol that eliminates transaction fees and can handle millions of transactions per second. For enterprise-level DApps, the EOS BC architecture allows both vertical and horizontal scalability. High TPS, low latency, high TPS, enhanced parallel performance, and high sequential performance are all promising aspects of EOS. Inter-BC communication is also supported by EOS (IBC). The consensus algorithm used by EOS is delegate proof-of-stake (DPoS).
- IoT Chain: It is a platform designed to provide a light system for IoT devices' security and scalability demands. To achieve lightning-fast performance, IoT Chain combines a directed acyclic network with the practical Byzantine fault tolerance (PBFT) consensus method.
- IoTeX: It is a BC-in-BC platform for M2M transactions that require privacy; the architecture consists of a public Rootchain and Subchains, which are managed by the Rootchain. Subchains are made up of either private or public BC and are responsible for controlling groups of linked devices. Subchains communicate with one another via establishing cross-BC transactions with the Rootchain.
- HDAC: IoT contracts and M2M transactions are handled via a Multichain platform. The consensus method for this platform is ePoW, an energy-efficient variant of PoW.
- Atonomi: Atonomi is a platform built on the Ethereum framework that provides immutable identity management services to help develop safe, trustworthy IoT devices.
- Hydrachain: It is an Ethereum platform extension that allows constructing a private ledger.

2.3. Blockchain Consensus Mechanism

The BC uses a consensus-based data update method. The newly produced data must be confirmed by all or most of the BC nodes before they can be transferred into the public ledger shared by all BC nodes [59]. Any node may read, write, validate data, and generate consensus in this distributed system, earning rewards according to one of four classic consensus mechanisms (i.e., PoW, PoS, DPoS, or PBFT) [60]. PoW, PoS, and DPoS vary in terms of who has jurisdiction privileges. In PoW (one of the earliest and costliest mechanisms, commonly used by Bitcoin and Ethereum), all nodes compete for the privileges on a level playing field based on their processing capacity [61]. In current BCs, PoW is the most widely used consensus method. The chance of mining a block in PoW is determined by the miners. PoS requires fewer processing resources than PoW, making it an energy-efficient method. The nodes in the PoS network are thought to be less interested in hacking (attacking) it [42]. Miners (nodes) must, thus, confirm their involvement in the form of cash regularly. As a result, the BC is ruled by the wealthiest players (miners), which appears to be unjust [42]. Stakeholders choose the delegates who manage the permits for authentication and accounting in DPoS. Bitshares' bedrock is DPoS-based. Proof-of-activity (PoA) is a hybrid technique that blends PoW and PoS. Instead of computational power, the proof-of-capacity (PoC) makes use of the available hard drive space. Permacoin, SpaceMint, and Burstcoin all use the PoC approach. PBFT is a byzantine fault tolerance replication algorithm developed by IBM, which mines each block grounded on runtimes within stable execution environments using a random option of manager. One BC network that uses PBFT is HLF. The architecture of consensus structures faces a challenge in terms of scalability. A variety of adapted protocols have been developed to meet the various specifications of BC applications.

Private BCs have unique characteristics, such as a smaller number of members and the ability to be semi-reliable, compared to public BCs. They are normally given a series of permissions as they are added to the system. As a result, these structures necessitate complex consensus processes that match these characteristics [56]. Paxos, grounded on state machine repetition and developed by Lamport and Microsoft as a distributed blocking service, are some of these alternative mechanisms. These methods have the advantage of being adaptations of formal algorithms, which means their characteristics have been scientifically proven [46]. RAFT, which separates core elements of consensus including leader selection, security, and record replication, applies a higher level of conformity to minimize the number of states to consider [46]. SIEVE treats the BC as a black box that compares the performance of each replica. The procedure is not validated if there are divergences between the replicas. The federated Byzantine agreement (FBA) is another PBFT version. Each participant in FBA keeps a list of trustworthy contributors and waits for these contributors to approve a transaction to be validated [62].

2.4. Related Work

This section explores a summary of the most recent and relevant surveys that addressed the challenges of BIoT integration. A summary of the previous survey studies is shown in Table 4. These findings and other studies' findings are discussed in great detail in Sections 3 and 4. Fernández-Caramés and Fraga-Lamas [4] discussed smart city BIoT applications. Ferrag et al. [63] provided an overview of existing BIoT applications, protocols, applications, privacy, and security considerations, as well as future research goals. In Reyna et al. [46], the possible benefits and approaches to integrating BC with IoT, as well as the associated advantages and BC platforms, were highlighted. Ali et al. [25] explored BIoT systems, as well as prospective applications and challenges. The application areas of BC in the industrial IoT (IIoT), as well as industry-specific obstacles and outstanding topics, were highlighted by [64,65]. Wei et al. [66] discussed the security risks associated with the confluence of BC and IoT, as well as possible remedies. Xie et al. [26] discussed the use of BC technology in numerous applications in smart cities, as well as the associated research challenges. Yang et al. [36] investigated the literature on merging BC and edge

computing systems; they reviewed challenges and highlighted numerous key components of this integration, including motivations, frameworks, enabling functions, and challenges. Ahmed et al. [67] provided a thorough evaluation of the research on utilizing BC. Ferrag et al. [68] studied BC-IoT protocols and offered several threat models and difficulties in BIoT networks. Rao et al. [69] assessed some of the most promising IoT applications and presented ways for overcoming key challenges, all of which should lead to successful IoT integration. In Wang et al. [70], the authors explored how BC can play the function of security enabler in IIoT, and they outlined the security needs of IIoT. Tseng et al. [71] simulated how network quality and system dynamics affect consistency using multiple realistic consistency models for Bitcoin Backbone Protocol (BBP)-based databases (BBP is carried out by players who construct a BC on the basis of the Bitcoin source code) [72].

Table 4. BIoT integration challenges—survey studies.

Recent Survey Article	Year	Domain	BIoT Challenges
Fernández-Caramés and Fraga-Lamas [4]	2018	BIoT applications	Technical, standardization, BC infrastructure, regulations, and different BC varieties.
Ferrag et al. [63]	2018	BIoT integration	Security attacks, adequate security framework, power consumption, trust management, BC infrastructure, skyline query processing, and vehicular cloud advertisement dissemination.
Reyna et al. [46]	2018	BIoT integration	Scalability, security, privacy, smart contracts, legislations, and consensus.
Ali et al. [25]	2018	BIoT applications	Scalability, security, privacy, resources, public/private implementation, big data and machine learning (ML), SDN network, and cellular network.
Alladi et al. [64]	2019	BC applications in IIoT	Scalability, security, privacy, energy and cost, resources, and regulations.
Wei et al. [66]	2019	BIoT integration security challenges	Security, data privacy, authentication and identity management, trust establishment, decentralized cooperation, and consensus protocol.
Xie et al. [26]	2019	BIoT for smart cities	Security, privacy, energy, incentive and punishment mechanisms, cost, and regulations.
Yang et al. [36]	2019	BC in edge computing	Scalability, security, privacy, artificial intelligence (AI), self-organization, function integration, resource management, and big data.
Ahmed et al. [67]	2020	BIoT in smart cities	Security, privacy, data, structure, bandwidth, and latency.
Ferrag et al. [68]	2020	BIoT for green agriculture	Scalability, ML and database of intrusion detection, choosing consensus algorithm, cryptographic protocol, security attacks, and slicing threat of 5G.
Rao and Clarke [69]	2020	BIoT integration	Security, privacy, computation and storage, the granularity of transactions, trust, and successful pilots of BIoT, and awareness.
Tseng et al. [71]	2020	BIoT-based database	Bitcoin Backbone Protocol (BBP) database.
Wang et al. [70]	2020	BC for industrial IoT	Performance, privacy, standardization, and complexity.
Al Sadawi et al. [11]	2021	BIoT integration	Scalability, security, privacy, resources, consensus choice, big data, device mobility, smart contracts, and standardization.
Bhushan et al. [73]	2021	BIoT unification	Scalability, IoT resources, BC infrastructure, BIoT and cellular network, privacy, and ML, and big data.
Farahani et al. [74]	2021	BIoT integration	Security, privacy, throughput, latency, resources, usability, and centralization.

Table 4. *Cont.*

Recent Survey Article	Year	Domain	BIoT Challenges
Majeed et al. [57]	2021	BIoT for smart cities	Scalability, security, privacy, sustainability, consensus algorithm, latency, processing and storage, and smart contract immutability.
Singh et al. [75]	2021	BIoT integration security challenges	Combined and zero attacks, infrastructure, and security requirements (key exchange, resources, performance, and threat management).
Uddin et al. [58]	2021	BIoT challenges and solutions	Scalability, security, privacy, connectivity, big data, and throughput.
Da Xu et al. [76]	2021	BIoT security challenges and recommendations	Performance, consensus, bandwidth, communication, block recording, integration with edge computing, and interoperability with 6G.
Yaqoob et al. [77]	2021	BC for healthcare data management	Scalability, regulations, interoperability, irreversibility, tokenization, integration with eHealth, data accuracy, and adoption culture.
Abdelmaboud et al. [8]	2022	BIoT integration	Scalability, security, identity management, interoperability, consensus related challenges.
Kumar et al. [78]	2022	BC and IIoT integration	Complexity, security, privacy, interoperability, heterogeneity, resources.
Yu et al. [79]	2022	BIoT security challenges in smart cities	Resources and power consumption, confidentiality, standardization, and the need for more studies.
Pennino et al. [53]	2022	BC as enabler of IoT economy	Scalability, transaction cost, unneeded functionalities, and computational power.
Alkhateeb et al. [80]		Hypride BC for IoT	Portability, resources, interoperability, computational power, and scalability.
This survey		BIoT integration challenges	An in-depth comprehensive discussion of BIoT's current challenges. Seven challenge categories including 28 challenges were identified. Discussion of best practices currently recommended. Discussion of BIoT's future challenges. Six main themes of future challenges were identified.

Several articles on BIoT challenges were recently published. Al Sadawi et al. [11] outlined the primary challenges that BIoT systems face and suggested a role in addressing them. They presented a two-layer BIoT architecture based on dew and cloudlet computing. Bhushan et al. [73] provided an overview of how to adapt BC to specific IoT requirements to construct BIoT applications. In Farahani et al. [74], the authors provided a comprehensive reference architecture, as well as the basics, current achievements, promises, and challenges of BC and IoT convergence. Focusing on consensus techniques and BC platforms, the important prerequisites for integrating BC with smart cities were discussed in [57]. Singh et al. [75] discussed the BC important elements, and they presented a full examination of potential security challenges, as well as the current solutions that may be used as countermeasures. Uddin et al. [58] reviewed the BC for IoT, fog, and cloud in the context of smart cities, eHealth, and intelligent transportation. Da Xu et al. [76] examined the state-of-the-art BIoT security applications, focusing on security features, challenges, technologies, techniques, and situations. Yaqoob et al. [77] discussed how leveraging BC for healthcare data management systems may encourage innovations and bring big gains, as well as the open research challenges impeding the effective implementation of BC in the healthcare industry. Abdelmaboud et al. [8] surveyed the literature about BIoT benefits, challenges, and frameworks. Kumar et al. [78] surveyed the literature about BC and IIoT integration. Yu et al. [79] investigated the challenges and disturbances of BIoT, the applications, and the future of BIoT-based smart cities. Recent reviews by Pennino et al. [53] and Alkhateeb et al. [80] highlighted

the most pressing issues with BIoT integration. Scalability, resources, and processing power were noted in both articles as the major challenges to BIoT integration.

This survey provides a comprehensive analysis and synthesis of all BIoT challenges identified in the literature. While the above survey studies reported some BIoT challenges and these challenges were based on certain BIoT applications, this survey provides a holistic overview of all possible challenges from all available BIoT applications. For instance, Da Xu et al. [76] and Singh et al. [75] provided a review of BIoT security and privacy challenges only, Yu et al. [75] provided a review on BIoT security in smart cities, and Uddin et al. [58] focused on smart cities, eHealth, and intelligent transportation sectors in their review. Since BIoT integration is still in its early stage, more studies are required to provide an in-depth understanding of the challenges and requirements of this integration [11,19,37,74]. By 2030, the IoT is projected to grow to 29 billion devices [81]. To effectively leverage the dispersed model and global potential of the IoT to adopt BC, further research in this field should be conducted [74].

3. RQ 1 and RQ 2—Current Challenges and Recommendations to Enhance BIoT Integration

This section answers the first and second research questions (i.e., RQ1: What are the challenges that face BIoT integration? RQ2: What recommendations have been provided in literature to overcome these challenges?). We looked through the pertinent literature to get the answers to the research questions. The search strategy used all well-known databases, including IEEE, MDPI, and Elsevier. This paper only contains academic papers that were authored in English. The full text, abstract, and title of the chosen papers were then examined. The study was excluded if it did not demonstrate any focus on IoT and BC. Table 5 displays the search terms, databases, exclusion criteria, and publications that were ultimately chosen.

Table 5. Selected article process.

Search Terms	Database	Inclusion Criteria	Selected Articles
(Internet of things OR IoT) AND (Blockchain OR Bitcoin OR Ethereum OR Multichain OR distributed ledger OR Cryptocoin OR Hyperledger Fabric)	IEEE Xplore, Elsevier ScienceDirect, MDPI Online, Google Scholar, Wiley Online Library, SpringerLink, SAGE Publication, ACM Digital Library, and Emerald Insight	Only papers written in English. Result: 517 papers.	[4–6,8–13,19,21,24–27,29,30,33,35,36,39,42,44–49,51,53,55–58,61–64,66–71,73–80,82–151]
		Papers excluded on the basis of the abstract (if not focused on IoT and BC). Result: 273 papers.	
		Papers excluded on the basis of the full-text evaluation (if challenges of IoT and BC integration not reported). Result: 122 papers	

Although BC technology can be one of the most appealing options to deal with the security and privacy difficulties in IoT, many researchers and academics have suggested numerous research papers that promoted the notion of BIoT integration. Most IoT security and privacy frameworks are centralized and, therefore, not well suited for IoT due to the difficulty of scale, traffic, and single point of failure [83]. Moreover, the danger of device spoofing, erroneous authentication, and decreased data exchange reliability are still facing BIoT [21,84]. In this section, a comprehensive taxonomy of the current BIoT challenges, as well as recommendations to mitigate the effects of these challenges, are discussed. It is worth noting that the majority of the research focused on Bitcoin's application in the IoT context; therefore, these challenges are concentrated on Bitcoin. Despite the ongoing debate about Bitcoin's applicability in an IoT setting, this study only reports on an assessment of the existing literature in the field's top academic databases.

Seven challenge themes were synthesized from the literature: security, privacy, technical consideration, scalability, computational processing, regulations and guidelines, and BIoT

design. Under each of these categories, several challenges are discussed. The taxonomy of current BIoT challenges is shown in Figure 2. According to the current research, these BIoT integration issues are mostly based on the Bitcoin BC disadvantages. Moreover, a lack of BC capabilities is the primary cause of regulations and standards, as well as BIoT design difficulties. On the other hand, both IoT and BC concerns affect privacy, security, computational processing, technical considerations, and scalability. These problems will have an influence on the performance of BIoT integration in any of these scenarios. Accordingly, it is essential to investigate the overall impact of all of these difficulties on BIoT integration.

Figure 2. BIoT integration current challenges.

3.1. BC-Based Security Issues

The security model incorporates confidentiality (the capacity to keep data hidden from unauthorized users), integrity (the consistency and correctness of data throughout its life cycle), and availability (i.e., ensures that data are available to the users at a required range of performance in any situation); it is regarded as one of the most critical concepts for ensuring security in any form [21,141,152]. Furthermore, security challenges associated with the nature of BC including BC security vulnerabilities, smart contract vulnerabilities, trust management, and authentication (i.e., using personal identification to validate requests)

are other concepts that are required for implementing policy, and limiting access should be discussed [21,141,153].

Although all BCs utilize cryptographic protocols to protect their data and activities, this does not mean that they are without flaws [68,153]. Accordingly, for heterogeneous devices, a multilayer security architecture must be developed. Before any amenities are delivered to users, the framework should first adapt to current resources and make judgments about the security methods to apply at the IoT levels. Intelligence is required for such an adjustable security framework that is susceptible to resource standardization for deployment in IoT systems [85]. In this section, we look at how employing BC in IoT applications might jeopardize the security concepts and requirements mentioned above. In addition, the recommended solutions are discussed. Table 6 highlights the solutions recommended for addressing security challenges.

Table 6. Recommendations for reducing the negative impact of security challenges.

Category	Challenge	Best Practice
Security	Confidentiality [26,44,86]	• Tradeoff between decentralization and authorization • Mixing techniques (e.g., private and public platforms) • MW technique
	Integrity [76,87]	• Hierarchical BC design • Message identification and authentication
	Availability [76]	• All countermeasures should be examined • Data storage, theft, cleaning, and destruction techniques
	Authentication [57,75,76]	• Combining smart contracts with lightweight encryption methods • SSI and DID techniques • Message identification and authentication
	Vulnerabilities [10,88,89]	• Elliptic Curve Encryption, attribute-based signatures, digital certificates, timestamps for uniqueness, cryptographic key sizes • Secure channels, separation of transactions, low priority to newcomers • Multi-chaincode access control on a hierarchical BC
	Trust [27,66]	• Globally consistent reputation assessment model • BC digitization

3.1.1. Confidentiality

Cryptography and encryption technologies are commonly used in private BC to ensure confidentiality. Allowing any network member to access, write, audit, and maintain the blocks is a standard approach to ensure availability in public BC. Only preselected nodes can read or write in private BC. Transactions can only be performed by participants or preselected nodes who hold the private keys for authentication [113]. Furthermore, the devices that will be used as nodes may present a confidentiality challenge. These node devices may vary in terms of security policies implemented due to limitations in specifications, capabilities, manufacturer restrictions, or challenges in the configurations and programming [63,154]. Multichain enables integrated management of user permissions to ensure that only selected participants may access transactions, restrict the types of transactions that are authorized (authorization ensures that the user has permission to conduct a certain activity), and mine new blocks safely without the need for PoW and associated expenses. BCs of the private and consortium types aid in the solution of this problem, but they restrict user access and limit the degree of decentralization. As a result, for certain use-cases, an optimal tradeoff should be implemented [44].

Because transaction pattern analysis may be used to establish the identities of users or devices secured by public keys, it is difficult to reveal an IoT private transaction history in a public BC. Entities must completely grasp their privacy demands to evaluate if a private or hybrid BC is best suited to their needs [86]. A BC technique known as Mimblewimble should also be discussed, which is a method of organizing and storing transactions in such a way that all transactions appear to an outsider to be random data. Only the participants

in the transaction have access to the data. As a result, Mimblewimble is also associated with the notion of confidential transactions [26].

3.1.2. Integrity

A BC registry is particularly useful in keeping data on a distributed infrastructure that is very resistant to physical damage [77]. However, because of computing, communication, and service capabilities, the quantity of data created in the IoT varies by location and device. IoT sensors are often basic and cannot be successfully monitored for an extended period. This increases security concerns. Furthermore, terminal devices seldom employ security features; as a result, attackers may simply gain passwords and other identifying information, which they can then use to publish and disseminate false information [76]. Moreover, IoT devices often deliver data straight to the control center in a wireless manner after data collection. However, without adequate shielding, this technique is immediately exposed. This will result in the collection of a huge quantity of basic data, as well as the disclosure of personal and private data. It even poses problems with social and public safety [87].

BCs are notable for their resilience and irreversibility. Because of the system's irreversible and immutable nature, smart contract code bugs are highly dangerous. Moreover, because BC data are immutable, new technology such as quantum computing potentially jeopardizes public BCs with encrypted data [10]. Data or transactions that have been saved on the BC cannot be easily changed. Although BC assures authenticity, it cannot guarantee data correctness. If the saved document, for example, includes erroneous or wrong information, it will waste resources. The BC is a tamper-proof ledger, but if an error occurs, the loss may be irreversible. This differs from a centralized solution, which may roll back the database in the event of an error [155,156]. This problem appears to be tough to tackle at the moment; it is based on increasing the degree of centralization. This, however, is subject to the underlying application context. Due to the need for monitoring in many circumstances, the criteria should not be entirely decentralized [27].

As a consequence, the IoT data record and block sequence are synchronized. Inconsistent (forked) BCs can originate from incomplete or inconsistent records [76]. One solution to the inconsistency problem is the hierarchical design of the BC [76]. The data from the whole network are verified and stored in the higher main BC. To accomplish block management, the lower sub-BC controls data for particular physical locations or device components and blocks BC data for distinct levels [76]. Moreover, one of the most often used authentication techniques in IoT is message identification and authentication, which is a protective mechanism for ensuring the integrity and security of data when transferring data [76].

3.1.3. Availability

Availability and consistency in BIoT are tightly linked together in a reversal relationship. If the availability is increased, then the consistency will be decreased, and vice versa. For example, Ethereum is considered faster than Bitcoin as shown clearly in the quicker block time, whereas it is less secure as the process of adding mined blocks requires multiple confirmations by many BC applications to prevent transactions from a double-spending attack, which happens when an attacker obtains service or commodities from the account holder, and then manages to restructure the transaction ledger so that the crediting transaction is reversed [75,112]. Moreover, BIoT is exposed to a 51% attack which will cause the availability to be controlled by the attackers. Among the main attacks is the distributed denial of service (DDoS) which can abuse the process of authentication and trust by harming the system with an out-of-control number of requests to update and insert new records [4,124].

During information transfer, sophisticated security procedures are frequently ignored. As a result, attackers have the chance to intrude while data are being sent. Once this occurs, it will have an impact on the IoT system, leading user rights to be violated and the system to

fail to execute specific duties [76]. The IoT captures substantial data and stores substantial personal and private information about users, such as passwords and personal preferences. All countermeasures should be examined about the stored data, including strategies to prevent data theft and destruction, as well as ways to prepare in advance if the aforesaid circumstance arises. Furthermore, the IoT system would appear to malfunction if data are maliciously combined with "dirty" data. As a result, it is also important to think about dealing with data cleaning and other events [76].

3.1.4. Authentication

Authentication challenges in BIoT range from the increase in storage size and computing power overhead to openness against certain types of attacks [68]. In the IoT, public key infrastructures (PKIs) make authentication and identity management easier. At the moment, the most widely used PKIs are certificate authorities and a web of trust grounded on somewhat excellent privacy [119]. However, because the growing number of devices would necessitate significant processing and storage requirements to allow for continuous message exchange and confirmation of identity, these traditional solutions cannot fulfill identity management needs [75].

To resolve these challenges, combining smart contracts with lightweight encryption methods to handle identities automatically is an interesting research proposal. We may develop specific smart contracts that contain [75] (1) a device recording containing device type, expiry date, public key, and so on, (2) information updating when a new IoT device enters the network that may include firmware upgrade and expiration date, and (3) obsolescence of devices that may include device registration, identity verification, and information update. Furthermore, by synchronizing the data stored in the BC, the full node may synchronize the state of identity-related smart contracts to offer identity authentication. Similarly, the lightweight node may successfully authenticate additional IoT devices after querying the complete node [75]. The decentralized identity management strategy, which makes use of BC immutability and automated smart contract execution, not only avoids identity fraud but also lowers the cost of developing trust in IoT systems when compared to CA-based techniques [75].

The user's identification in several services (e.g., smart city) is currently given via central authority-managed digital identity management systems. Self-sovereign identity (SSI) and decentralized identity (DID) allow people to fully manage their digital identities without the need for a third-party intermediary. This gives consumers control over how their identifiable data are shared. BC-enabled IoT services, SSI, and DID can be used for decentralized authentication and authorization. DID and SSI, on the other hand, present several difficulties, including human dependence such as losing the private user's key [57]. Identity identification and message authentication are two types of network authentication. To ensure dependability, identity authentication requires a key. One of the most often used authentication techniques in IoT is message identification and authentication, which is a protective mechanism for ensuring the integrity and security of data when transferring data [76].

3.1.5. Vulnerabilities

Mining vulnerabilities: The integration of BC with IoT is exposed to miners' attacks. These attackers' miners may control the BC if they dominated the power of the consensus mechanism [85]. The attacker (miner) keeps the mined blocks in a private branch instead of broadcasting them until it becomes longer than the public chain and then releases them once to force the BC to change the public chain and gain the rewards of this process [18,112]. This kind of attack will cause a waste of resources and may affect the performance of the fog network that uses BC [122,157]. BIoT applications that use smart contracts are exposed to certain types of software vulnerabilities which can be used by attackers to harm the architecture [34,50,74].

Attacks: Although using BC technology enhances the security of IoT applications, BIoT integration is still exposed to certain types of security vulnerabilities. BIoT applications are open to the application-dependent and application-free of attacks [36]. BioT is exposed to the majority attack (also called 51% attack) which will cause the availability to be controlled by the attackers (e.g., Ethereum and Bitcoin) [88,158]. Several attacks have been identified in the literature and are explained as follows:

(1) The DDoS attack can abuse the process of authentication and trust by harming the system with an out-of-control number of requests to update and insert new records [4]. Latency and, in some cases, the use of low-performance devices in the BIoT applications will make this architecture exposed to race attacks [85].

(2) The smart contract attack occurs when a customer uses the same cryptocurrency for several transactions. In a PoW-based BC, this type of attack is particularly simple to execute since the attacker may take advantage of the interval between the commencement and confirmation of two transactions to start an attack rapidly [88].

(3) Border gateway mechanism (BGP) hijacking is another attack. BGP is a de facto directing mechanism that controls the delivery of IP packets to their final terminus. Attackers either use or modify BGP routing to intercept BC's network traffic [88]. Because of the extreme concentration of some Bitcoin mining pools, BGP hijacking will have a significant impact if they are targeted. The attackers can essentially divide the Bitcoin network or slow down block propagation.

(4) Manipulation attacks (i.e., unlawfully intercepting, modifying, or deleting sensitive data while they are being sent or stored) include four types: the eclipse, overlay, man-in-the-middle (MiTM), and tampering attack. (A) The attacker can use the eclipse attack to monopolize the target's outgoing and incoming networks, thereby separating the victim from the others in the network [88,143]. The attacker can then alter the victim's awareness of the BC or allow the victim to leftover computational resources on outdated perceptions of the BC. In addition, the attacker can use the victim's computational capacity to carry out its harmful operations [10,159]. (B) The overlay attack, which utilizes the receiver's public key, makes use of BC flaws to maliciously wrap an encrypted quantity to a novel transaction [10]. Addressing this attack can be achieved by verifying the timestamps. As a result, diverse inputs underneath the same dealer can be detected and linked to several transactions. (C) MiTM attacks take advantage of flaws such as private key leaks to spoof two parties' identities and surreptitiously interrupt and manipulate their communications. Some BC frameworks, including Ethereum and Bitcoin, are still vulnerable to this attack [10]. (D) In the tampering attack, the attackers try to change the signed transactions that are being distributed in the network, such as the addresses and other data, before propagating them to the P2P network for validation [10]. Bitcoin, Litecoin, and Monero are PoW cryptocurrency-based ledgers that are particularly vulnerable to this attack.

(5) Identity-based attacks: The adversary's goal here is to create a false identity, pose as a genuine user, and obtain access to and influence the targeted system. Several attacks can be listed under identity-based attacks [10,75]. (1) Replay attacks are designed to spoof two parties' identities, interrupt their data, and repeat them to their intended terminuses. Such attacks come as a result of the disorientation that certain nodes may suffer during a soft or a hard fork, and they are frequently carried out via vulnerable cryptocurrency-based protocols. (2) Key attacks (which make use of flaws in key schemes) allow unauthorized users to control the identities of the nodes that are participating through improper usage or storage of the keys. (3) In Sybil attacks, the attacker uses leaked keys to build a large number of false identities that may serve as authenticating nodes and undertake malicious transactions to boost or decrease the reputation level of the nodes that are being targeted. (4) In impersonation attacks, opponents can use weak or leaked private keys to impersonate genuine users and undertake unauthorized operations in the system.

(6) Whitewashing attacks: In these attacks, nodes with a bad reputation take advantage of several system flaws to re-enter the BC with new identities. There is currently no formal solution to this assault; instead, TrustChain gives nodes with new identities lesser priorities and capabilities [10].
(7) Quantum computing may be viewed as a danger to Bitcoin, since the computational capacity of these machines may be sufficient to compromise the integrity of digital signatures. It will just take a few minutes for a brute-force attack to crack the encryption and get the encryption keys [10]. Furthermore, technology advances with time, and new vulnerabilities and security flaws are discovered every day [46].
(8) The liveness attack is another attack described in the literature that allows a low-mining-power attacker to interrupt communications between subgroups with equivalent mining power for a short period [160]. The balance attack against PoW-based BC is another attack [161], in which an attacker can temporarily disrupt communications between subgroups.

3.1.6. Trust Management

While actual collaboration is required to complete specific services and operations, establishing trustworthy cooperation and confidence in an IoT system is difficult as each node can join and exit the system at any moment, and nodes' identities can be anonymous [66,126]. Accordingly, trust management is among the key challenges when adopting BIoT as IoT needs to connect hundreds of devices to form P2P data delivering and sharing [12,85]. The following are some of the trust difficulties that have been found in the literature:

(1) Trusting the software updates for fog node devices and trusting the owners of those devices are among the most common challenges for IoT device permissions and communications when using BC [68,82].
(2) Another challenge related to trust is granularity (i.e., resolution of the measured and recorded physical quantity), which is the amount of confidence that should be placed in the party providing the service and the extent to which the service receiver to pay for is trusted. For example, if an electric grid service provider wishes to be paid for every watt of electricity provided as it is provided, this may place undue stress on the system [107].
(3) Another related problem is that it is difficult to implement real transaction validation. For example, a service provider could send out 10 W of electricity, but the receiver might only report 9 W. What is the best way to deal with this collision? The meters must be calibrated. Then, an independent method of determining the quantity of power sent and received is required [69].
(4) Lastly, because of the trust difficulties, rating agencies may be able to give information on trust. This is similar to how online marketplaces like eBay employ ratings for sellers and customers. The resolution of trust takes place outside of the BC environment. As a result, while sending and receiving a specific quantity of Bitcoin is guaranteed, the service it may represent is not [69].

To mitigate the challenge of trust, a global reputation assessment model based on BC technology to improve the efficacy of the above trust models can be used [66]. Establishing a node assessment mechanism grounded on irreversible block data is one such option. First, with BC technology, any node in the block can obtain the behavior data of other nodes. The reputation of the nodes would be computed automatically after inserting their activity data into the reputation evaluation algorithm. When calculating reputation, the kind of behavior and timeliness should be taken into account, since these two factors represent the node's performance over time and impact the ultimate reputation distribution [66]. As a result, while choosing the appropriate sort of behavior and timescale, it is critical to consider the nodes' motivations and preferences to effectively govern node behavior and develop trust [66,67]. Moreover, since the BC is unable to validate external data, it must be digitized to achieve its full worth. There are two types of solutions that might be considered. One is the addition of a third party who can vouch for the accuracy of the input data. However,

part of the decentralization feature may be sacrificed as a result. Another option is to use innovative approaches to tackle asset digitization challenges. This may need the creation of IoT devices capable of enabling exclusive identification [27].

3.1.7. Learned Lessons

As this study discovered, BIoT-based applications may face a variety of security challenges, including multiple threats, trust, data availability, confidentiality, and integrity. To address these challenges, numerous interesting solutions have been developed, including leveraging a combination of private and public BCs, integrating smart contracts with lightweight encryption approaches, and message identification and authentication. While some of these techniques may provide solutions to current challenges, these solutions may not be enough if quantum computing takes place, for instance. Other solutions, such as the global reputation model [28] to ensure trust, seem to be very hard to achieve due to the heterogeneity of IoT devices and standards.

3.2. BC-Based Privacy

Similar to security, the privacy model contains various goals and requirements that must be accomplished to protect privacy, including identity privacy (i.e., hidden user identification from unauthorized users), data privacy (i.e., data is only accessible to authorized users), location privacy (i.e., unauthorized parties are unaware of the user's location), and usage privacy (i.e., unauthorized parties are unaware of use patterns) [84,162,163]. Below, four challenges of BIoT privacy preservation are discussed: data privacy, identity privacy, location privacy, and user privacy. Table 7 highlights the solutions recommended for addressing privacy challenges.

Table 7. Recommendations for reducing the negative impact of privacy challenges.

Category	Challenge	Best Practice
Privacy	Identity [10,26,46]	• Change address approach, Zerocash, Zerocoin, and Monero in Bitcoin • Zcash and zkSNARK in Ethereum • Identity control service in HLF • User permissions in Multichain • Techniques such as ring and blind signatures, pseudonymization, homomorphic commitments, and zero-knowledge proof
	Data [10,25,46]	• Hawk, DHT, and splitting data in Bitcoin • Encryption in Ethereum • User permissions in Multichain • Identity control service in HLF • Off-chain data saving • Confidential transactions, homomorphic encryption, BC segregation
	Location [10,57]	• BC segregation • Stealth addresses (one-time address) • Pseudonymization technique
	Usage [10,26]	• BC segregation • Cryptographic commitment methods and Ouroboros Crypsinous BC

3.2.1. Identity Privacy

When it comes to privacy, anonymity is a huge worry for BC platforms, and it may be a key deciding factor in which platform to choose. The fundamental challenge is that all transactions are openly documented and accessible to anybody, on public platforms [164]. The selected anonymity strategy fails if the transactions can be connected to their owners or if the owners' identities are revealed [127]. The owners of transactions should not be identified by any participants in the network in an ideal scenario [62].

BIoT shares transactions with other parties and will identify its users by public or hash key, which means that, if attackers analyze the traffic and transactions of the BC, they will

be able to identify the actual identity of the users [112]. Furthermore, the devices that will be used as nodes also present another privacy challenge. These node devices may vary in terms of security policies implemented due to limitations in specifications, capabilities, manufacturer restrictions, or challenges in the configurations and programming [63].

Existing systems have used a variety of tactics with varying degrees of anonymity. Because the identity of the users is unknown and should not become known, public BCs necessitate a higher level of privacy [46]. To improve the amount of anonymity, Bitcoin uses the change address approach, which involves using various addresses for various transactions [113]. Zerocash and Zerocoin, two popular proposals to address Bitcoin's anonymity problem, suggest that Bitcoin extensions have unidentified transactions, obscuring the sender, recipient, and information itself. On the other hand, to make transactions undetectable, Monero uses a ring of signatures., making it impossible to link them to a specific person or machine [46]. The Ethereum team is working with Zcash to implement a zero-knowledge succinct non-interactive argument of knowledge (zkSNARK) transaction mechanism. The method allows you to conceal a transaction and keep it fully confidential [62]. HLF offers a control service for identity and access using private channels to enable privacy control on BC networks. Users may regulate and restrict access to their shared information in the network [133]. Members of the network are identified by their public identities; however, they are not required to be aware of the information exchanged in the network [46].

Pseudonymization is the dispensation of personal data in such a way that it can no longer be linked to a specific person without the help of supplementary information. Supplementary data are stored separately and are subject to technological and organizational safeguards to prevent personal data from being linked to an identified or identifiable natural person. [10]. In most cases, pseudonymization is accomplished by substituting user identities with pseudonyms, i.e., particular form identifiers that do not enable the re-identification of persons on their own. Because each wallet user is assigned a unique address that appears to be random (i.e., produced by cryptographic procedures and serving as a pseudonym), BC cryptocurrency platforms such as Ethereum and Bitcoin adopt pseudonymization procedures.

Other approaches to improve anonymity include blind signatures, homomorphic commitments, ring signatures, composite signatures, mixing services, and zero-knowledge proof [26]. To commit data without exposing them to others, homomorphic commitments employ homomorphic encryption. Confidential transactions leverage homomorphic promises to mask transaction amounts. Before signing the message, the message owner blindfolds the content using a blinding factor, which is a type of digital signature. Because they allow a person to sign a communication without understanding what it contains, blind signatures are commonly employed in privacy-related applications. A ring signature is a digital signature used to sign a message's content by a group member. Identifying the actual signing member of a group is computationally impossible. The ring signature used in the Monero platform combines confidential transactions with ring signature and produces ring confidential transactions. A composite signature is made up of several separate signatures that are not in any particular order. Individual signatures might be difficult to compute from a composite signature. Composite signatures can be used to make Bitcoin-like currency more anonymous. Mixing services received from several clients can also be used for anonymization. It is impossible to track the activities of users using the mixing service [26].

3.2.2. Data Privacy

One of the most important characteristics of BC technology is its immutability. As a result, it is critical to ensure that data, such as healthcare data, are correct [77,165]. Although BC is considered everlasting storage, in reality, some cases decline this claim. For example, in 2014, some hackers were able to steal almost eight million Vericoins from the cryptocurrency exchange platform called MintPal [4]. Wallet apps are one source of

vulnerabilities in Bitcoin that might expose transaction data. Ethereum's data and contracts are encoded but not encrypted.

HLF devotes a significant chunk of its protocol to addressing security concerns such as preventing transactions from being connected to users, access control techniques, and digital signatures. However, not all of these functions have yet been implemented. Data in the public BC can be encrypted to improve privacy. Hawk keeps track of transactions that are encrypted [46]. The Hawk compiler is in charge of transforming programmers' generic code into cryptographic primitives that allow for transactional information anonymity. In addition to encryption, the Enigma project divides data into unidentifiable bits and distributes them over the network in such a manner that no node ever gets access. It stores data references in a decentralized off-chain distributed hash-table (DHT) (decentralized storage that searches for information using pairs of keys and each node is accountable for several keys) that is accessible via the BC [46]. Because private BCs must offer authentication and authorization procedures by definition, the challenge of data privacy can be addressed in different ways. In Ethereum, cryptography is used to limit the exposure of sensitive data and data segmentation to improve data privacy. Multichain has user permissions built in to limit perceptibility and set limits on allowed and miners.

Another way to protect data privacy is to store sensitive information outside the BC (i.e., off-chain) [46]. Because it would be impossible to store significant amounts of data inside the BC, this method benefits systems that manage big volumes of data [26]. Furthermore, they are well suited to systems that handle extremely sensitive data and require stricter access controls, such as healthcare applications. In this method, the public BC can store newscaster data, allowing evidence of data integrity and time stamps to be provided. Users may examine the BC to verify data without authorities, and data are safely kept outside. These off-chain sources should not be a source of single-point failure and should be fault-tolerant [46,118].

Other approaches to improve data privacy include confidential transactions, homomorphic encryption, and BC segregation [10]. In confidential transactions, several solutions are available to implement this strategy on a BC platform (i.e., hiding data from unauthorized third parties while still allowing transactions to be completed) such as Besu (an Apache 2.0 licensed open-source enterprise Ethereum) in Hyperledger and Parity (open-source-based Ethereum application) in Ethereum. Homomorphic encryption might be used to transfer and conduct operations on data without disclosing their secret values, thus embodying the aforementioned concept of secret transactions in some way. In Ethereum and Quorum, privacy techniques based on homomorphic encryption are available. There is no common ledger with all transactions visible to all members in BC segregation; rather, just a portion of these transactions are accessible to all members [10]. As a result, unauthorized third parties are unable to learn of the presence of transactions that they should not be aware of. Several platforms, including HLF, EOSIO, Corda, and Hyperledger Iroha, provide privacy methods based on BC segregation. Only a preset and identifiable group of members share in a given communication in Corda, while the rest of the network is oblivious to the transaction [10].

3.2.3. Location Privacy

Pseudonymization can help in hiding the location of users. However, to avoid complications emerging from the use of a single pseudonym, a user may be connected with many pseudonyms (or addresses) to avoid transactions relating to a single user from being linked [57]. Utilizing one-time addresses for this purpose will rapidly result in a huge number of addresses per participant. Bitcoin, Ethereum, Monero, and other cryptocurrencies include privacy measures based on one-time addresses [10]. The usage of so-called stealth addresses, which were created for financial BC platforms to ensure payee anonymity, is a good example of BC technology. The payer produces a one-time address for each transaction with a single payee by suitably applying cryptographic processes. This technique is used in Bitcoin and can be applied by other platforms, such as Ethereum and Monero [10]. Moreover, the BC segregation technique can be used to ensure location privacy [10].

3.2.4. Usage Privacy

Although each BC user has a public pseudonymous address, all transactional data such as receiver and sender are accessible by all participants. Users' activity can be monitored by examining the data saved on the BC. Real users' identities might be discovered by merging the information analyzed from the BC with specific external data. When a user's identity is disclosed, all of the user's actions will be tracked, and their personal data, including financial secrets, will be revealed [26].

As for location privacy, the BC segregation technique can help in maintaining usage privacy [10]. Litecoin announced that MW would be included to improve privacy. Cryptographic commitment methods, such as the Pedersen commitment, provide another approach to obfuscate transaction information. By using these methods, a sender can commit transaction data and communicate the commitment rather than the data [26]. The sender cannot later mislead about a fictitious value of the original data (which remains hidden). Third parties may be able to verify that a transaction's input and output numbers are similar without disclosing them in financial transactions if such guarantees are made (e.g., cryptocurrencies). Other cryptographic promises, such as the Ouroboros Crypsinous BC, are also utilized [26].

3.2.5. Learned Lessons

Privacy preservation should be monitored against four entities; identity, data, location, and usage privacies. While the majority of the previous efforts in this context focused on addressing identity and data privacies in IoT-based applications, location, and usage privacies have not been paid enough attention. Decentralized access control incorporated in smart contracts, block headers, or even the BC's transactions have been offered as possible options to handle the privacy of BIoT-based apps. Another promising solution is by deploying a tiered architecture, in which isolated private BCs are linked by a public BC. However, data integrity within private BCs remains an issue in a tiered architecture. In BC, preventing double-spending and ensuring auditability necessitates the sacrifice of anonymity. While current explorations have suggested interesting answers, protecting privacy in public IoT-based applications continues to be a research difficulty.

3.3. BC-Based Technical Considerations

Unlike conventional BC approaches that are generally applied to a dispersed system (e.g., the bandwidth, consensus algorithms, and cryptographic mechanisms), physical features ground the IoT devices' communication. The communication between the nodes will be uneven if the communication quality is inadequate. This will result in issues such as the inability to validate the block and the waste of computer resources, thereby destroying the BC system's consistency [50,76]. The creation rate of BCs and the capacity of blocks, on the other hand, are quite high because of the huge number of IoT devices and transactions. Although BC can improve IoT security, its implementation in IoT is incompatible with BC's limited block capacity and channel propagation [90]. As a result, the use of BC improves the security of IoT. However, in practice, some technical problems exist including consensus algorithms, smart contracts, unstable communication, network bandwidth, cryptographic mechanisms, and cryptographic keys [138]. Table 8 highlights the solutions recommended addressing technical challenges.

3.3.1. Consensus Algorithms

Even though the PoW mining method overcomes the problem of transaction reliability, it wastes substantial resources. Mining incentives can also result in extremely concentrated mining pools. With just seven transactions per second, the decentralized architecture provides a long-term consensus period [38]. The BFT method is used to create the consensus mechanism in the consortium chain, which overcomes the problem of the PoW algorithm's low efficiency and excessive resource consumption. For complete P2P communication monitoring of abnormal activity, the BFT algorithm offers a high number of signature

confirmations [166]. However, because of the high communication complexity, the system has substantial overhead, which affects consensus competence. This falls well short of BC's expectations for IoT security. As a result, to satisfy the development of BIoT requirements, consensus methods must be updated and enhanced [57,76].

Table 8. Recommendations for reducing the negative impact of technical challenges.

Category	Challenge	Best Practice
Technical considerations	Consensus algorithms [57,74,76,90]	• Consensus algorithm built on game theory • Consensus algorithm built on AI • Adaptive nature consensus algorithms
	Smart contracts [10,11,46,57]	• Symbolic execution, theorem proving, abstract interpretation, and model checking • Sperate data and logic; call delegation from proxy contract to logic contract
	Communication [13,44,76]	• Edge and fog computing (FC), side-chains, notary systems, hash locking, and standardization • Inter-BC communication protocols
	Heterogeneous devices [11,76,77]	• Multilayer security framework • Standardization of resources
	Bandwidth [76]	• Edge computing and FC
	Cryptography mechanisms [44,88]	• Tradeoff between consensus mechanism used based on the application • New consensus mechanisms required
	Cryptography keys [4,68,88]	• AES for low-power secure communications • Use of ciphers such as Simon • Further study and empirical assessments are still needed

In most cases, IoT devices communicate tasks to gateways that can complete the work. Off-chain solutions may increase functionality by transmitting data outside the chain to reduce latency [8]. While several projects are aimed at incorporating complete BC nodes into IoT devices, mining remains a major stumbling block in the IoT because of its restrictions. The IoT is largely made up of devices with limited resources, yet it has tremendous processing capacity on a global scale. Babelchain, for example, adapts PoW for IoT applications using a new consensus method named proof of understanding (PoU). PoU, rather than hiring miners to decode hash problems, translates from other protocols, requiring fewer energy resources [74]. While tackling important difficulties in IoT connectivity, the effort is mainly focused on practical processing. Rather than agreeing on transaction status, participants agree on the communication's meaning (i.e., format, action, and content).

Alzahrani et al. [90] proposed a consensus method based on game theory. A variable number of validators is decided dynamically on the basis of game theory in their consensus method. The chance of danger is reduced by selecting just a legitimate number of honest validators. Chen et al. [109] proposed an AI-based consensus method to combine the benefits of PoW, PoS, and DPoS. Chaudhry and Yousaf [167] discussed designing consensus mechanisms for different applications including BC type, scalability, fault tolerance, throughput, and bandwidth. However, all these solutions are yet to be further investigated [57].

3.3.2. Smart Contracts

As BC-based programs, smart contracts enable the automatic storing of legally binding agreements on the BC. This feature enables a BC to go from a certain application area, such as financial transactions, to the management of a larger variety of transactions and assets [62,108]. Smart contracts, on the other hand, are vulnerable to several attacks, including timestamp (changing the block timestamp value to the node's maximum future time limit) and transaction-ordering dependencies (when the block's timestamp is used for certain contract decisions, but timestamps may be modified by miners), reentrancy vulnera-

bility (when a function calls an untrusted contract before resolving any consequences), and Mishan (where the transaction may be reversed if the exceptions made were not correctly managed) [88]. Furthermore, contract execution by computers has various disadvantages since it exposes them to technology issues such as hackers, viruses, and communication failures [46]. In addition, smart contracts may become overburdened in situations when several data sources are necessary. Smart contracts are distributed and decentralized; however, they do not spread performing tasks to manage enormous numbers of tasks [11].

Because of the serious consequences that might emerge from smart contracts' vulnerability, academics have developed several approaches and tools to detect these flaws. To strengthen the security of smart contracts, model inspection, symbolic accomplishment, and theorem proving can be employed [10]. Theorem proving is the most common way to formalize smart contracts, which is mathematically modeled while the desired qualities to be demonstrated are expressed. Due to its appropriateness for any system that can be described mathematically, theorem proving may be regarded as an adaptable verification approach. Symbolic execution, which is a software testing approach that aids in the development of data and evidence of a program's quality, is a delicate solution in which several execution pathways are methodically evaluated at the same time, without the need for real inputs. Symbolic execution has led to significant advances in several important software dependability applications, and it is the most widely used methodology for detecting vulnerabilities in EVM bytecode. Model inspection, which is an automated formal verification methodology that applies to systems that may be described using a finite-state model, deploys model checking software, such as NuSMV, to verify the results. Model inspection is most commonly used with Solidity language, although it may also be used to validate contracts written in other languages.

Smart contracts are irreversible and aid in the establishment of trust between contractual parties. However, even in the event of flaws, vulnerabilities, or new business-logic specifications, a smart contract's code is typically not upgradeable. The modified code of the smart contract is usually deployed in a new case with a new contract address, which might cause difficulties [57]. One technique to upgrade a smart contract is to delegate calls from the proxy contract to the new logic contract. The data are held by the proxy contract, while the logic contract performs the new logic. For each change, the logic contract identity is restructured in the proxy contract [57].

3.3.3. Unstable Communication

BC is built on the assumption of steady network connections that is not viable for IoT, which frequently has poor network IoT device connectivity or network instability owing to node failure. In most situations, the state of IoT devices cannot be determined until they are tested, while, in others, the devices function normally for a length of time before changing due to a variety of factors such as disconnection, short-circuit, and program obsolescence [11]. IoT devices, unlike computer nodes on the Internet, are influenced by the transfer of physical underlying data. Wireless access allows IoT devices to connect, such as distributed drones and automobile network connections. If the wireless link's quality is inadequate, the node's information transfer will become unstable, which results in a change in the BC network's topology [76]. Some device information would be unavailable to the verification node, which would lead to incomplete data processing. Because of the enormous number of interconnections and broadcasts between nodes, broadcast storms can quickly occur, consuming too much network capacity. As a result, the IoT network's performance may be impaired, if not completely shut down.

Edge computing, as a form of cloud computing, can handle dispersed node communication. The challenge of network bandwidth requirements may arise [76]. Side-chains, notary systems, hash locking, and standardization are all possible answers [44]. Moreover, inter-BC communication protocols can be utilized to address projects that are focused on interoperability [44]. Limiting the consensus over various network sections or linking nu-

merous BC by establishing inter-BC communication represents two more viable BC-based solutions [13].

3.3.4. IoT-Based Heterogeneous Devices

IoT devices are available in a wide range of sizes and designs; they operate on several operating systems and, of course, have varying performance characteristics. The variety of devices adds another layer of difficulty to the BIoT integration [85]. As low-cost, low-power devices grow increasingly common, the BIoT integration may become increasingly vulnerable to hardware flaws. Furthermore, because of the large degree of heterogeneity across IoT devices, the original BC architecture, which just employs an "address" to describe a node, is unsuitable for IoT devices [75]. Moreover, multilevel security systems must be created for heterogeneous devices [11,76]. Before delivering services to users, the system should first adapt to current resources and make judgments about the security methods to apply at the IoT levels. Intelligence is required for such a dynamic system [77].

3.3.5. IoT-Based Network Bandwidth

The use of BC has grown in popularity, as has the number of nodes. Only some nodes are coupled in a P2P architectural network, and the present network can meet bandwidth needs. In the future, the number of nodes in BC will rapidly expand. Because of the enormous number of interconnections and broadcasts between nodes, broadcast storms can form quickly, consuming too much network capacity. As a result, the IoT network's performance may be impaired, if not completely shut down [76]. Edge computing can handle dispersed node communication. The challenge of network bandwidth needs for large-scale deployment of BCs is expected to be overcome in IoT [76].

3.3.6. Cryptographic Mechanism

The PoW algorithm is utilized to build the consensus process in the public BC. Although the PoW mining method overcomes the problem of transaction steadiness, it wastes many resources. Mining incentives can also result in extremely concentrated mining pools. With just seven transactions per second, the decentralized architecture provides a long-term consensus period [76]. The BFT algorithm is used to create the consensus mechanism in the BC consortium, which overcomes the problem of the PoW method's low efficiency and excessive resource consumption. For complete P2P communication monitoring of aberrant activity, the BFT algorithm offers a high number of signature verifications. However, communication complexity is substantial, resulting in system overhead [76].

When employing BC, the private key is considered as the IoT device's identification and security credential, produced and managed by the IoT device rather than third-party organizations. Because the Elliptic Curve Digital Signing Algorithm cannot create enough arbitrariness, it can lead to a vulnerability where an attacker may obtain the user's private key [88]. The user's private key cannot be restored after it has been lost. If the user's private key is obtained by attackers, the user's BC account is vulnerable to tampering. Because the BC is not reliant on any centralized third-party entities, it is impossible to follow the criminal's activities and retrieve the changed BC information if the user's private key is taken.

Ethereum and Bitcoin, as the top BC platforms, are not well suited to meet the demands of a large number of users, since they can only guarantee a modest maximum number of transactions. As a result, sharding techniques, as well as off-chain and side-chain techniques, have already demonstrated a high level of potential for addressing scalability issues. For BC applications, privacy is also a sensitive and crucial subject. All transaction data are publicly accessible in public BCs; nevertheless, openness in BC must be balanced with the security of personal and sensitive data. BCs of the private and consortium types alleviate this problem; however, they limit user access, limiting decentralization. The security degree of BC is determined by the consensus technique employed. For example, PoW-based consensus is subject to a 51% attack, but BFT-based consensus is vulnerable to a 33% attack. Small BCs with fewer users are more vulnerable to these attacks, whereas

large BCs can provide far greater security but suffer from power centralization and hashing. As a result, for certain use-cases, an optimal tradeoff should be implemented. The BC community is working on improving and adapting consensus procedures to make them more efficient and secure [44].

3.3.7. Cryptographic Keys

To provide privacy and security in a BC, public-key cryptography is required. However, the limited resources IoT devices fail to meet the computational requirements of contemporary safe cryptographic methods. When used on IoT devices, asymmetric cryptography based on RSA is particularly sluggish and energy-intensive. As a result, not only should the computing load and memory be considered when selecting a cryptographic system, but also the amount of energy spent [68]. Most IoT devices cannot use the current RSA key sizes. Since the failure of 768 bit and 1024 bit RSA implementations in 2010, a 2048-bit key is a minimum size considered safe. Although it is conceivable, using a 2048 bit certificate with an ephemeral key exchange technique imposes significant overhead and computation requirements that are impossible to meet on the restricted hardware [4,63].

Elliptic Curve Cryptography (ECC) (where cryptographic keys can be generated more efficiently, smaller, and quicker using Elliptic Curve Theory-based PKI), on the other hand, is a significantly lighter substitute to RSA. ECC has previously been demonstrated to beat the power consumption and speed of RSA when used on resource-limited systems [4]. Hash functions are particularly important in BC since they are used to sign transactions. As a result, hash functions must be safe in IoT applications (i.e., they must not produce collisions), be quick, and spend as little energy as feasible [4]. SHA-256d, SHA-256, and Scrypt (an algorithm that makes big attacks on the hardware more expensive by requiring substantial memory) are the most common BC hash algorithms (used by Litecoin, Gridcoin, and Dogecoin) [88]. SHA-256's performance has been tested on a variety of IoT devices, including wearables. Researchers that looked at the footprint and energy need of SHA-256 in ASICs concluded that using Advanced Encryption Standard (AES) is more efficient. Other researchers recommended utilizing ciphers such as Simon because of the power restrictions, but more research and practical evaluations of real-world BIoT applications are still required [4,88].

3.3.8. Learned Lessons

According to our study, various elements can cause technical challenges, including communication and bandwidth, cryptographic mechanisms and algorithms, heterogeneity of IoT devices, and smart contract architecture. Despite several solutions provided to address these challenges, mining remains a major issue in the IoT due to its limitations. A smart contract code is typically not upgradeable; furthermore, more BC nodes will increase exponentially in the near future limiting the bandwidth, and BC is built on the assumption of steady connections, which is not feasible for IoT, which frequently has poor network connections.

3.4. BC-Based Scalability

Scalability refers to the capacity of BC's IoT environments to operate without sacrificing QoS as the number of IoT devices grows [57]. BC-based transactions, unlike standard transactions, are implemented in every single node and ledger. Shared ledgers will become safer and more scalable for data storage if the BC is implemented. However, numerous challenges must be addressed, including the ledger storage facility, limited technological progress, a qualified staff, a lack of standards, time fluctuations and processing speed, computing capacity, and scalability challenges [75,107]. An interesting research topic in BC-based systems is storage [26]. Scalability in the context of BCs relates to several factors, including storage, block size, and transaction cost. Table 9 highlights the solutions recommended addressing scalability challenges.

Table 9. Recommendations for reducing the negative impact of scalability challenges.

Category	Challenge	Best Practice
Scalability	Storage [8,57,74]	• Filter, compress, or normalize IoT data • Parallel mining using PoW and horizontal scalability • On-node and off-node computing storage • Encrypt data and upload in decentralized off-chain storage such as IPFS
	Block size [62,64,91]	• Increase the block size • Use variable-length blocks
	Cost [26,28,62]	• Combining numerous transactions • Less expensive protocols may be employed • Simulating and evaluating the BC-based system • IOTA framework might help to overcome this problem in part

3.4.1. Storage

Storage is one of the main issues of the integrated architecture between IoT and BC. For instance, to add a new node, all the previously used nodes need to store the complete chain to validate the new node because the underlying BC framework is constantly becoming larger, which makes it a complex and error-prone process [69]. Every member in the BC network keeps a local copy of the entire distributed ledger. When a new block is verified, it is broadcast over the whole P2P network, and each node adds the approved block to its ledger. This places a strain on the storage capacity available to IoT devices. If 1000 users trade a single 2 MB image every day in a BC application, a BC node would require around 730 GB annually [58]. As a result, when BC works with IoT data, the difficulty is to manage the growing data storage requirements. Moreover, unlike traditional paradigms, cryptographic algorithms must be constrained to function within the restricted storage capacity of IoT devices. The storage requirements for broadcasts necessary for exchanging keys must be met to ensure effective employment of communication protocols and security for IoT [85]. One of the consequences of a BC-enabled environment is that accounting ledgers may need to be kept for an infinite period. This necessitates offloading accounting data from IoT devices to local or distant servers in the case of IoT devices. Although storage prices are continually decreasing, solution providers must select where and when these data are kept [69].

Filtering, compressing, and normalizing IoT data have been recommended using several approaches. Saving data supplied by the IoT is useful on many levels since it covers target services, communication, and embedded devices. Data compression reduces the amount of time it takes to process, transmit, and store created data. Furthermore, as demonstrated by Bitcoin-NG, BC's consensus mechanism, which generates bottlenecks, may be changed to increase bandwidth and minimize transaction latency, leading to better IoT transitions [74]. Another concept is to merge the BC with the current P2P storage, which enables storing huge amounts of data off the chain, to alleviate the storage problem. Using the DHT, the method creates off-chain storage. The raw data are kept on the DHT off-chain, with just the data references remaining on the BC. The SHA-256 hash is used as a reference. The InterPlanetary File System (IPFS) is a P2P dispersed file system that combines principles from earlier P2P systems such as BitTorrent protocol, DHT, Git, and self-certified filesystems. Filecoin acts as an incentive layer on top of IPFS to provide a completely distributed file storage system [26]. Desema, a decentralized service marketplace system based on Ethereum and IPFS, has been presented. Service metadata and big data are kept off-chain in the Desema system, with Ethereum merely storing data references [26].

3.4.2. Block Size

The amount of storage and bandwidth available in BC is increasing all the time. This is due to the log's increasing size and the increasing quantity of data it holds [68,88]. A freshly created transaction is validated by adding it to a block in the BC. The size of the block might impact the amount of time it takes to insert and validate data [11]. When it

comes to the block size, the current platforms have taken different approaches. Bitcoin is one of the systems with the lowest block sizes, with block sizes capped at 1 MB [64]. Any block that exceeds this restriction is considered illegitimate. This restriction impacts the maximum number of transactions that may be included in each block, resulting in competition between transactions, favoring those with greater fees [106]. Multichain has increased this restriction to 32 MB block sizes, and they aim to expand it much more in the future [62]. Other platforms, like Ethereum and HLF, have chosen to use variable-length blocks [91].

3.4.3. Cost and Transaction Fees

Public BCs are more expensive when it comes to transaction costs. Bitcoin, for example, is thought to have a rather high transaction cost. This is hardly surprising, given that the nodes participating in the process of reaching a distributed consensus must be compensated financially. Ethereum's costs are smaller than Bitcoin's, yet they still add up to a significant amount. A user can reduce fees by combining numerous transactions into one bigger transaction. Because the nodes in permissioned BCs know each other, obtaining a distributed consensus is not as CPU-intensive; alternatively, less expensive protocols may be employed. In fact, in the vast majority of situations, the costs may be agreed upon in advance by the participants [62].

When it comes to design, the cost is a delicate topic. The cost is made up of two parts: design and operation. The cost of implementing and running a BC-based system, on the other hand, remains unknown [26]. Other than Bitcoin BC, there are few BC systems in full production at the moment. It is impossible to estimate the cost of implementing and running a system based on BC at scale. As a result, focused studies to determine the possible cost of a BC-based system are required. Simulating and evaluating the BC-based system such as NYUAD, City of Things, and SmartSantander is one option [26]. High costs for IoT systems are inconvenient. Tangle, an IOTA framework, might help to overcome this problem in part [28].

3.4.4. Learned Lessons

According to this research, the three parameters that determine the scalability of BIoT are storage, transaction cost, and block size. Several solutions have been offered to overcome scalability difficulties, but the off-chain technique may be the most promising solution, mainly for off-chain computation and storage in FC, where a big number of resources is dispersed and can give sufficient capacity for devices to conduct compute-intensive jobs in real-time. FC, on the other hand, does not include infinite storage. That is, because of the massive volume of data generated by BIoT applications, FC will be insufficient to handle it, at least in its current form and capabilities.

3.5. Computational Processing

Although IoT devices' processing capabilities are improving, participating as a node capable of contributing a transaction to a BC is still computationally demanding. IoT devices have limited resources. These devices often have limited computational capabilities, weak network connectivity, low battery capacity, and limited storage. BCs, on the other hand, have their unique set of criteria. Because BC data are large, it is impossible to keep the entire BC in each IoT device, especially since the IoT creates vast amounts of data in real-time, exacerbating the problem [69]. In most circumstances, the devices function normally for a length of time before changing due to a variety of factors such as disconnection, short circuits, and program obsolescence. The following discussion goes over these challenges. Most IoT devices are short of the computing capacity needed to perform the computations required to interact with BC directly [74]. Computational processing in the context of BCs relates to several factors, time latency, power, and transaction throughput. Table 10 highlights the solutions recommended to addressing computational challenges.

Table 10. Recommendations for reducing the negative impact of computational challenges.

Category	Challenge	Best Practice
Computational	Time [57,74,92]	• Data chunking or the detection of indicators • Propagation scheme using closest neighbor selection (CNS) • An acknowledgment-based scheme
	Power [26,57]	• Renewable power sources • Energy collecting • lightweight consensus mechanisms or algorithms
	Throughput [26,93]	• Off-chain transactions and sharding • Reducing the block interval time • TDAG-based systems

Resource management is a significant concern for the IoT. A matching contract for resource provisioning is utilized in this distributed integration system to couple a resource request with a resource offer. It identifies the characteristics of computing resources related to CPU type, disk space, and RAM, and it executes the pairing with various regulations. Add-on devices that provide specific processing capabilities are increasingly becoming accessible, even if the IoT device itself does not have the requisite computational capability [69]. The Intel Movidius neural computing stick, for example, integrates deep neural networks in hardware for tasks such as filtering and object identification. Moreover, feasible solutions will include data chunking or the detection of indicators such as departures from predicted thresholds. More computational power is necessary for IoT or edge-of-network devices to accomplish such processing [131]. Due to resource limits, consensus mechanisms must be redesigned to be energy-efficient and lightweight, as well as improved energy collecting strategies [85].

3.5.1. Time

Latency refers to the length of transaction processing time [11]. The growing number of nodes and transactions may result in an increase in the mining and validating block's time. Even while public BCs can help speed up the insertion of a new block, it is still challenging to add blocks at the pace that IoT data are created [69]. The latency limitation has a substantial impact on scalability capabilities [69]. In addition to processing at local nodes, data transmission across nodes contributes to the latency. The latency of the BC network is determined by both of these delays. Forking is caused by latency caused by propagation delays. A miner who is not a broadcasting miner may broadcast their block in a network before receiving another miner's block. Miners can own several blocks as a result of the forking effect [57]. In healthcare, authenticating a block takes around 10 min, which might compromise security countermeasures during that time. Because any delay might affect exam analysis, healthcare systems must be flexible and available at all times [69]. BC transactions need time to be processed, which will lead to extra latency once applied in BIoT [75].

Feasible solutions may include data chunking or the detection of indicators such as departures from predicted thresholds. Propagation delay must be kept to a minimum to avoid the forking effect. The closest neighbor selection (CNS) technique was utilized in [168] to decrease the propagation latency in the BC network. Another method of avoiding the forking impact is to utilize acknowledgment when a new block is received, to signal whether or not forking has happened [92]. Following the fork, the process of block generation is restarted. This procedure repeats itself until the block is updated without forking. Because there is a chance of a rollback if a lengthy chain does not include the necessary block in the future, the forking effect produces probabilistic confirmation of transactions [57].

3.5.2. Power

Computational power is necessary for IoT devices to execute the consensus process. BC is built on the assumption of steady network connections, which frequently have weak

network IoT device connections or an unstable network owing to node failure due to battery life, for instance. Moreover, the consensus technique needs a large amount of computational power, which costs energy, making it impractical for low-power IoT devices [11]. Mining devices' hardware necessitates substantial computing power, which raises the cost of implementation and limits the design options for device kinds. For example, the PoW is deemed costly in terms of computing power and storage requirements [68]. According to Ghosh Bouri [169], up to January 2022, the worldwide Bitcoin yearly electricity consumption was about 130 TWh (terawatt-hours of energy), equating to around 75 megatons of carbon dioxide ($MtCO_2$) annual emissions. High computational power is necessary to keep transaction data encrypted, which consumes substantial electricity power [4]. Despite the highly promising attempts to use different algorithms, these algorithms are still in their early stages. Other algorithms have limitations that some applications cannot tolerate; this may motivate the research community to seek out other alternatives to the PoW algorithms while maintaining the excellent security and dependability that PoW provides. As a result, studying energy-efficient consensus techniques for BC systems is intriguing [26,78].

3.5.3. Throughput

Throughput refers to the maximum processed number of transactions in a given amount of time. The throughput is calculated by dividing the number of transactions per second by the number of concurrent workloads and IoT nodes. IoT gadgets create terabytes of data in real time, but BC is not intended to store that much data [11]. Only a few transactions per second are processed by some BCs such as Bitcoin. This is undeniably a constraint for BIoT systems. Implementing consortiums or private BCs may help in this case [11]. Quick synchronization of new Blocks across BC nodes necessitates more bandwidth, which can enhance BC throughput. As a result, increasing BC throughput due to numerous transactions in IoT systems is a difficulty [58]. The BC implementation architecture, on the other hand, needs the execution of a huge number of transactions each minute. This makes BIoT integration challenging in either way.

Although Bitcoin is one of the most popular platforms, it is also one of the least scalable, due to the PoW algorithm and blocks size limitations, which result in a low transaction rate [25]. While Ethereum's smart contracts, which allow for the execution of sophisticated logic, enable a wide range of transaction types and block sizes, transaction validation still takes a long time [93]. Public permissioned systems can manage significantly greater transaction rates; HLF can handle 100,000 transactions per second [36]. In terms of transaction rates, private BCs offer no benefit over public permissioned BCs. Multichain, on the other hand, has the benefit of being Bitcoin-compatible [62,107]. As a result, appropriate schemes must be carefully devised to enhance the throughput of BC systems while keeping sufficient security to handle billions of IoT devices and sustain a large number of real-world transactions. Many ways to enhance the throughput of BC systems have now been proposed [26,28,93,170,171].

- Segregated witness, commonly known as SegWit, is where digital signatures are separated from the rest of the transactions and moved to the end of the block. As a result, transaction sizes are smaller, and one block may carry more transactions.
- Off-chain transactions: Here, if nodes make several transactions, off-chain micropayment channels are formed between them to carry out serval transactions off the chain rapidly, and BC only processes the final payment transaction.
- Sharding is a useful method for enhancing the horizontal scalability of BC systems. Nodes are partitioned into shards using BC sharding. Only a tiny percentage of all transactions are processed by each shard. Transactions are handled in parallel in this manner. Two examples of sharding BC systems are Elastico and OmniLedger.
- Reduced block interval time: Block generation in BC systems consists of two processes: transaction serialization and leader selection. The selection of one or more leader nodes is the responsibility of the leader election. Transaction serialization refers to the validation of transactions and the generation of new blocks by the chosen leader nodes.

Leader nodes are picked at a low pace to reduce collisions during leader elections. Every 10 min, for example, the Bitcoin BC leader node is chosen. Each leader election in a typical BC system can only create one new block. Transaction validation and block creation are delayed due to the connection of leader election and transaction serialization. Rapid transaction serialization and slow leader selection need to be separated to minimize block interval time and enhance throughput. Many alternatives, including ByzCoin, Bitcoin-NG, and Solida, have incorporated the concept.

- Systems based on TDAG: TDAG is the next step of BC's development. In a TDAG-based system, transactions are immediately added to a graph, creating a graph of transactions. IOTA is an example of a TDAG system. IOTA's underlying technology is Tangle. When a new transaction is added to the IOTA Tangle, it selects between two prior transactions to approve. When a transaction is authorized by a large number of other transactions, it is said to be confirmed. Because transactions are added quickly in blocks, IOTA outperforms traditional BC systems in terms of throughput.

3.5.4. Learned Lessons

According to this article, the three factors controlling the computational processing of BIoT applications are response time, power, and transaction throughput. Because of their considerable networking overheads and performance, BCs are limited in their applicability across constrained IoT devices. A possible approach is end-to-end communications over the BC, by employing computationally capable IoT gateways. However, the challenge in this manner is getting users to transmit their transactions to the gateways without depending on a centralized system. On the other hand, off-chain solutions may increase functionality by transmitting data outside the chain to reduce latency. Moreover, the collaborative design with FC may be useful, in which FC server can conduct the heavy operation and enhance the response time. Yet, privacy and security will be other issues of off-chain computational services.

3.6. Regulations and Guidelines

There is a lack of regulation when it comes to BC-related applications and operations. To ensure the continued and robust growth of the BC ecosystem, we need competent and consistent rules and regulations. Regulations in the context of BCs relate to several factors, including lack of standards, incentive and punishment mechanisms, and lack of awareness. Table 11 highlights the best practices for addressing guidelines and governance challenges.

Table 11. Recommendations for reducing the negative impact of regulations challenges.

Category	Challenge	Best Practice
Regulations	Standards [75,77]	• Use smart contracts to integrate laws and legislations into source code • Use existing standards in BC networks
	Incentive and punishment [26]	• Incentivization enhances nodes to participate in data verification • Punishment mechanisms are required for BC systems to avoid penalizing malevolent nodes and double-spending attacks • Nodes make a deposit to BC systems before they may create new blocks • Confirmation time
	Awareness [69,74]	• Provide materials that enhance awareness of BIoT

3.6.1. Lack of Standards

Regulators face several challenges as a result of BC's creative nature. The existing centralized regulatory structure, however, is incompatible with the BC decentralized model, particularly for public networks, because territorial rules are an issue. BC may have some illegal content data stored on this architecture, which may lead to legal complications for all the members of the architecture including the miners [70,71]. The smart contract used in BIoT does not follow any legal informants outside the transacting groups. Therefore, it

makes the integration exposed to many attacks that cannot be followed by law enforcement and will lead to conflicts among transacting groups. However, a government or another authority might use smart contracts to integrate laws and legislations into the source code to control block-related matters as part of a BC agreement [65].

Interoperability is critical for allowing communication across different BC networks. Although the lack of standards in BC benefits developers, it creates substantial communication challenges owing to a lack of interoperability. The availability of various BC networks with various consensus models, transaction processes, and smart contract functionality is a big barrier to interoperability [77]. Using existing standards in BC networks is one option for dealing with this problem. For example, in their BC-based supply chain operations, IBM and Microsoft use GS1-based data standards to ensure compatibility [77]. Another option is to create new standards [77]. The Enterprise Ethereum Alliance (EEA), for example, has released a standard version of the Ethereum BC.

The International Organization for Standardization (ISO) has established and developed several BC and distributed ledger projects since 2016 [34]. This investigation is only the beginning of the process of ensuring that the BC and the regulations are in sync. Both the industry authorities and the governments face difficulty in figuring out how to enhance the lawful conduct of BCs and associated technology. Japan, Singapore, and Switzerland are among the nations considering BC-friendly laws [44]. However, because each distributed ledger has no central management, worldwide standards should be developed. The European Union Parliament has previously passed a BC resolution titled "Distributed ledger technologies and BC: fostering trust via disintermediation". Furthermore, the International Association for Trusted Blockchain Applications (IATBA) was founded, bringing together BC vendors and users from around the world with representatives from governmental and standard-setting bodies. Therefore, policymakers can intervene in the BIoT industry by developing security and privacy norms and standards [75].

3.6.2. Incentive and Punishment Mechanisms

Different incentive strategies are employed to mine blocks in BC networks. PoW is used by some, whereas PoS is used by others. There are, however, several other algorithms. Mining rewards and the payout for executing a contract are the incentive mechanisms used in BCs. Choosing the right reward for the BC application is a delicate matter that has ramifications for nodes and miners [77]. To demonstrate the problem, the first miner who solves the puzzle of PoW will be paid a specific quantity of bitcoins in Bitcoin BC [11]. It may be presumed that nodes are self-interested; thus, incentive mechanisms are required to incentivize these nodes to donate their efforts to data verification. In cases where a collection of nodes generates blocks collectively, the allocation of rewards among these nodes must be properly structured [26]. Punishment mechanisms, on the other hand, are required for BC systems to avoid double-spending attacks and penalize malevolent nodes. The confirmation time is one approach, which means that a node's economic incentives can only be used after a substantial confirmation period. Once a poison transaction is discovered within the confirmation period, the malicious node's economic incentives will be nullified. Another option is to make use of the deposit. Nodes must deposit BC systems before they may create new blocks. The nodes are punished and lose a portion of their deposit if a poison transaction occurs. This method relies heavily on a proper deposit amount. If the deposit is too little, malicious nodes will be unaffected. Acting as a node to produce new blocks is costly if the deposit is too big, and a casual node is incapable of doing so, resulting in centralization. As a result, efficient reward and punishment systems must be properly devised to attract more companies and individuals to engage in BC-based smart cities [26].

3.6.3. Lack of Awareness

Nodes were not intended to locate one another since BC was not developed for the IoT from the start [74]. BIoT integration is affected by the level of knowledge and awareness

about both technologies, BC and IoT, which is very low in most of the domains, as the distance between what people in different domains know and the cutting edge of industrial technology continues to increase [151]. Furthermore, materials that provide an integrated picture of new technologies such as cloud computing, IoT, security, encryption, and BC are severely lacking. Accordingly, the National Security Agency in the United States has been financing efforts to provide teaching materials in cybersecurity-related subjects [69].

To allow the use of BC technology in the healthcare industry, a cultural transformation is necessary. The majority of existing healthcare systems are centralized and manual, making them vulnerable to data breaches and single points of failure. On the other hand, in the present BC scene, there are not enough individuals who are sufficiently educated or equipped to manage sophisticated P2P networks [115,172]. The demand for BC-related employment has risen by about 2000% in the past few years. Finding skilled coders, on the other hand, has become a major issue. Because BC technology is still in its infancy, it will take time for the development community to embrace it [77].

3.6.4. Learned Lessons

According to this article, the regulations and guidelines concerns of BIoT are characterized by a lack of standards, regulations, awareness, and incentive and punishment policies. Despite its security characteristics, BC use is restricted because of a lack of norms and regulations. For instance, attackers might use exploitable gaps in smart contracts to launch assaults like the DAO hack [25].

3.7. BIoT Design

IoT may benefit from BC technology in a variety of ways. However, because they were not designed specifically to support IoT contexts, different BC components must be tuned to make them suitable for such settings [4,108]. IoT applications create substantial traffic; thus, the design of BIoT apps needs to be tailored to handle it. In this section, we discuss the BIoT architectures and BC platforms that enhance BIoT integration. Table 12 highlights the solutions recommended for addressing design challenges.

Table 12. Recommendations for reducing the negative impact of design challenges.

Category	Challenge	Best Practice
Design	BIoT architectures [4,91,94,95]	• The completely distributed design is preferred for BIoT architectures; however, in cases of cost or power constraints, alternate techniques may be appropriate • The BC process should be converted to a more competent layer, such as the fog, edge, cloud layers, and SDN
	BC platforms [46,57]	• Tradeoff between BC platforms based on different applications • The character of different BC platforms is determined by their design goals

3.7.1. BIoT Architecture

The architecture supporting a BC needs to be tailored to the volume of traffic generated by IoT applications. Such an issue is more prevalent in classic cloud-based architecture, in which the node uses IoT gateways to send data to the cloud [113]. Traditional cloud-based IoT designs have several flaws such as the potential to disrupt the whole network by executing DoS attacks if only one device is compromised [112,173]. Traditional cloud-based IoT designs rely on the cloud; however, the degree of reliance varies greatly in practice. There are additional gateways that do more complex functions (for example, sensor fusion), although, in most cloud-based applications, the majority of the processing is conducted in the cloud [4,95]. The architectural issues connected with BIoT service providers have been examined in several designs suggested in the literature. Liao et al. [91] examined four architectures, including totally distributed, distributed things, fully centralized things, and faux distributed things. In most situations, the completely distributed design must be followed by BIoT architectures; however, in cases of cost or power constraints, alternate

techniques may be appropriate. Dorri et al. [166] presented a hypothetical lightweight architecture for BIoT applications that reduce the communication overhead caused by BC. The suggested architecture is separated into three layers: smart home layer, shared storage layer, and cloud layer for distant storage.

Regardless of the architecture described above, the BIoT design should take into account the limits and problems mentioned above. The BC-based theoretical architecture should consider delivering services and connecting heterogeneous IoT devices. This approach is based on a service discovery system that is built using multilayered hierarchical BC. Accordingly, the BC process should be converted to a more competent layer, such as the fog layer [22,128,139,142], edge and cloud layers [5,12,146], cloud layer [130], or SDN layer [30,94]. Li et al. [106] proposed a multilayered IoT design that aimed to reduce the complexity of BC deployment by establishing multiple layers in the IoT ecosystem and employing one BC at each level. Samaniego et al. [174] provided another solution to the challenge of hosting a BC on typical IoT hardware that is resource-restricted. It looked at how FC and cloud computing architectures may be used in BIoT applications. The FC systems beat cloud-based systems in terms of latency reaction time under heavy transmission loads, according to the system's empirical performance evaluation. Another edge-based computing architecture was proposed by [175], in which the IEC 61,499 standard was used to create distributed and hierarchical platforms. To govern the fog nodes, the authors of [30] recommended using SDN for BIoT applications. Cloud computing is used to conduct computationally heavy activities, and FC is used to give low-latency data access. According to the findings, the suggested design improves throughput, reduces delays, and identifies numerous real-time IoT network threats. In the fog layer, the authors of [142] developed a lightweight system on the basis of an SDN network. They developed a pBC network among the devices to solve the security problems in the centralized design of SDN.

3.7.2. Blockchain Platform

As discussed in Section 2.2 and Table 2, many features of BC platforms differ, such as the level of security, privacy, and scalability. For example, Bitcoin and Ethereum are not designed to achieve the demands of a high user number, since they can only guarantee a low maximum number of transactions [125,134]. Both Bitcoin and Ethereum are connected to a 51% attack level, whereas HLF and Multichain are tied to a 33% attack level. The first stage in recommending a BC-based solution for a project is to assess if a BC-based solution is feasible given the project's needs. Several elements are used to assess if a BC use case is valid [57] including the necessity for data to be exchanged among several parties without the need of a centralized middleman, the immutability of information, access permissions, the visibility and availability of shared history ledger-data to the participants, good transparency or audit trail for each participant, if the project involves a long-running, repetitive process that may be mechanized and coordinated using BC, and if the project's requirements are not met by the centralized ledger.

Moreover, the IoT's heterogeneity and limits should also be considered in smart contracts. Smart contracts should be used in conjunction with filtering and grouping techniques to enable apps to handle the IoT. A discovery technique might allow for device addition on the fly, enhancing the power of these apps. Lastly, smart contract-based actuation methods would allow for speedier IoT responses [46]. Accordingly, different varieties of BCs will need to be adjusted for different purposes. Furthermore, when customers wish to combine IoT devices and BC, the first step is to determine which BC best meets their needs. As a result, a technique for testing different BCs is required.

3.7.3. Learned Lessons

The BIoT architecture and BC platforms used, according to this article, characterize the BIoT design-related challenges. BC is a distributed design that adds new data using a distributed consensus mechanism. Since IoT devices produce huge levels of traffic, merely

employing public BCs in IoT applications is not a feasible option. Other authors suggested using FC and SDN to process BC including utilizing FC and SDN. Other suggestions included utilizing various BC platforms based on BIoT applications. As a result, different BC kinds were not altered for various applications. In addition, the first step in BIoT integration will be to figure out which BC is suitable for them. Hence, a method for evaluating various BCs is needed.

4. RQ 3—BIoT Integration: Future Concerns and Trends

We should point out that present research on utilizing BIoT applications is still in its early phases. The majority of the projects are proof-of-concept studies. Building benchmarks for these apps and having real-world, bigger implementations are both intriguing. Benchmarks can assist in the development of more efficient IoT applications utilizing BC [28]. The decentralization of BC technology is its distinguishing feature. The BC has far-reaching consequences for research and practice [149]. At the same time, BC's development and acceptance of the IoT have been fraught with several challenges. The first major unresolved challenge is how to assess BC performance. The performance of BC platforms varies depending on the number of nodes, consensus methods, network circumstances, and other factors. Finding ways to increase BC performance to fit IoT applications is also an outstanding topic after this knowledge [28].

Because the BIoT has received so much attention and has been studied so extensively, it may be impacted by a variety of different technologies. In the meanwhile, the BIoT influences them. We briefly explore the future concerns and trends in this section, as well as some larger thoughts on using BC in various technologies to encourage IoT development. These concerns and trends can be categorized into seven major categories: big data, new business model, AI, quantum computing, double-chained IoT, and interoperability. Table 13 summarizes the major future directions from the literature.

Table 13. Summary of future challenges and directions.

Challenge	Research Directions
Big data [11,25,26,68,70,73]	• Complex big data analytics methodologies on limited resources IoT devices are not directly possible • Providing authentication to the training datasets might be a huge difficulty • Promoting users to submit their data, using incentive mechanisms, in order to develop ML models
New business mode [76,82,96]	• Trust issues among diverse users • Applications that minimize response time and maximize resource usage
AI [8,26,36,97–99]	• Low-quality data can drastically damage the learning process • Access to the training dataset by hackers enables them to change attacks type
Quantum computing [77,82]	• Poses major security concerns for public-key cryptography-based BC systems • Quantum-resistant encryption development
Double-chained IoT [76]	• Security methods developed lack a credit guarantee and data security measures • Data BC and transaction BC can be integrated into a double-chain mode based on BC
Interoperability [8,78]	• Issues arise in data sharing: data exchange, data valuation, and data buildup • Multi-signature and BC shared ledger can be used to enhance interoperability
IoT, BC, and edge integration [12,105,142,150]	• Inclusion of edge computing or SDN to enhance the security and privacy • Edge and fog also have their own concerns that should be considered

4.1. Big Data

The study of IoT real-time produced data is becoming increasingly popular. This sort of data is typically diverse and large in volume, yet it has large value for a business [11]. Analysis of IoT big data might uncover hidden useful and relevant information that may help users make better decisions. The business demands of BC financial services,

for example, are significant, necessitating the addition of big data and related analytic capabilities to the ledger offered by the whole BC [117]. A group of more than 40 Japanese banks has inked a contract with Ripple to simplify the movement of payments between bank accounts and undertake real-time and low-cost transactions [70,145]. Another example was recently released by the Indian government by setting up a gene database system based on block linkages for 50 million people [70].

Despite these efforts, bringing traditional big data analysis to the IoT is a big challenge due to the following concerns [26,68,73]: (1) resources and computational capabilities are limited for IoT devices, which makes it impossible to use complex big data analytics methodologies on IoT devices directly. A potential alternative is to upload the data to clouds for processing and big data analysis, although this might result in significant latency and privacy problems; (2) the digital signature of a public/private key guarantees privacy in the BIoT. However, anonymous data may make decrypting and executing big data analysis a challenge and a time-consuming operation that leads to ineffective data analytics.

4.2. New Business Mode Based on Blockchain

IoT will link gadgets, in the future, to handle data exchanges. IoT-connected devices will have clever varied applications and functions to generate commercial value within the provided rule logic. To verify the legitimacy of transactions, it is required to accurately record transaction needs to be issued by IoT devices. Because the participants participating in such collaborations and transactions are usually from diverse stakeholder groups, establishing trust connections can be challenging using the traditional centralized architecture. As a result, IoT device collaboration and transactions can only take place inside the same trust domain. The application business value of IoT is substantially reduced as a result of this [76,96].

For trustworthy intermediates, BC can provide direct transactions. IBM and Samsung collaborated on Autonomous Distributed P2P Telemetry (ADEPT), a BC-based authentication solution. This technique creates a dispersed smart device network. ADEPT allows linked devices to interact securely and effectively, as well as execute sophisticated business logic. When a home's washing machine's supply of washing powder runs low, for example, the washing machine's smart contract control may order the powder directly from the supplier, and BC will validate the order and execute the payment transactions among the smart devices linked to the ADEPT [76]. Advanced IoT applications are needed for forecasting or traffic control. This will necessitate general application refactoring approaches based on quantum architectures and can be applied to cognitive models. These apps use a variety of edge resources to minimize response time while running and to maximize resource usage [82].

4.3. Artificial Intelligence

With a big volume of shared data of good quality and a strong BC-based distributed computing system, BC may make it feasible to move AI closer to the IoT [36,116]. To make intelligent judgments for improving automated chores such as scheduling and energy transactions, AI methods can aid IoT to a large extent [8,26]. These approaches can enhance edge computing scalability and increase the compute capacity for IoT data processing [97]. IoT devices run the learning models locally and communicate updates to several dispersed combination servers [176]. These servers exchange their related IoT devices learning models before performing global model aggregation [177]. There is no need for a centralized aggregation server, in this way. Several studies have used a federated learning model in IoT devices to address this issue. For example, Qu et al. [177] developed a BC-enabled federated learning method for FC. To provide resilience, this method aggregates local learning models in a dispersed way rather than using a unified server. All updates from local fog servers, in the end, are provided to all IoT devices that interact throughout the learning process. In another example, Savazzi et al. [176] proposed a federated learning strategy that relied on cooperation between IoT devices, without the need for a fog server,

to avoid the situation when IoT devices are unable to communicate with their fog server owing to communication restrictions.

Furthermore, by utilizing resource optimization strategies and scalable ML approaches, AI can assist in optimizing QoS and reducing breaches of services [82]. ML and BC have recently become increasingly popular. For example, Outchakoucht et al. [178] proposed a dynamic access control system based on BC, with access control strategies implemented as smart contracts. Smart contracts are updated based on reinforcement learning algorithms, which are utilized to improve and change access control rules. Roldan et al. [179] suggested combining ML with complex event processing techniques for real-time threat detection and effective event management.

However, the IoT application determines the federated local learning model type that should be utilized. According to [99], a feed-forward neural network (FNN) learning model is suitable for DDos attacks applications, a convolutional neural network (CNN) model suits smart car infotainment and image-based disease diagnosis applications, a support vector machine (SVM) model is suitable for detecting malware and slicing network applications, a long-short term memory (LSTM) model is suitable for edge caching and traffic prediction applications, and a K-means model suits IoT networks proactive caching and sensor network clustering applications. Because noisy or low-quality data can drastically damage the learning process, the algorithm requires a solid data sample to produce appropriate training datasets, which is not the case in BIoT [36,98]. Accordingly, decentralizing IoT with ML faces a significant barrier in terms of training dataset authenticity [68,102]. Furthermore, if the adversaries are conscious of the attacking nature and have access to the training dataset, this may easily change the attack or attack type [73,180]. As a result, determining the precise nature of an attack to properly discern between desired and undesirable network states is another difficult problem that requires additional exploration [25].

4.4. Quantum Computing

Although BC-based frameworks can provide edge computing systems with a greater level of security, super-scale data integrity necessitates fast hash computation. To overcome this challenge, quantum computers must be used to build adaptable systems to data properties [82]. Quantum dot architecture can help with the creation of new hashing and encryption mechanisms that make data integrity more difficult. To offer all stakeholders a safe environment that facilitates this paradigm change, the privacy and trust models established for serverless computing require special consideration, which might represent a challenge in the future [82]. Quantum computing is quickly increasing the processing capacity of contemporary computers, posing major security concerns for public-key cryptography-based BC systems. BC systems are predicated on the idea that traditional computers can swiftly decrypt large quantities. As a new technology, quantum computing can solve extremely complex matters very quickly. As a result, it poses a severe data security danger to BC technology. One proposed solution to this problem is to use quantum-resistant encryption instead of traditional digital signatures [77].

4.5. Double-Chained IoT Security Scheme

Studies on decentralization, data tampering, and information sharing have been conducted in light of the lack of credit guarantee methods and data security issues in IoT. A lightweight data-sharing security method for IoT is also a potential growth path [76]. To safeguard data gathering and transactions, the data BC and transaction BC can be integrated into a double-chain mode based on BC. The data BC, in particular, employs consensus algorithms to create a data ledger that prevents data tampering; in contrast, BC offers a decentralized, traceable, tamper-proof, and efficient ledger to enhance payment operations between IoT nodes [76].

4.6. Interoperability of IoT and Blockchain

This describes the exchange of data between various entities [78]. Resource sharing is a critical component of BIoT apps' compatibility. However, resource sharing confronts other obstacles, involving user identification, price, and resource management. Issues arise in data sharing as a result of data exchange, data valuation, and data buildup [76]. As a platform-centered BC solution, a safe mechanism with low cost can be straightforwardly deployed [8]. The BC approach's shared ledger increases the passage of real-time information among multiple groups to be tracked, lowering the cost of managing the shared data method. The network linked with power can benefit from a BC strategy [181]. The multi-signature energy trading system allows coworkers to retain transaction security and calculate energy costs while posting anonymously. It also addresses difficulties such as self-discipline management, interaction, accurate measurement, and optimization selections [8].

4.7. IoT, Blockchain, and Edge Computing Integration

Edge computing and software-defined networking (SDN) are two solutions offered in the recent literature to reduce IoT difficulties related to capacity and capabilities [12,142]. The concept is to transfer BC processes and consensus to the edge or SDN, which has additional capabilities and storage space. This appears to be a sensible problem-solving approach [102]. In the edge and SDN architecture, the BC system's nodes are all linked to the network, and each one keeps a copy of the transaction [105]. These data are often updated across all nodes to maintain their integrity. This enables a large number of nodes to make verification decisions based on prior transactions even while they do not trust one another. This improves security and privacy, as well as eliminates the need for centralized authentication authorities. In the SDN architecture, in order to provide an appropriate attack detection model, SDN continuously monitors and analyzes traffic data throughout the whole IoT network, whereas BC offers distributed detection mechanisms to address the single point of failure problem with the current design [142,150]. However, the scalability, lack of standards, and interoperability issues that these technologies (i.e., fog computing and edge computing) also compromise should be taken into account [162].

5. Discussion and Limitations

This paper reviewed the available literature to give readers a thorough grasp of the issues surrounding BIoT integration. We did this by responding to three study questions: What difficulties does BIoT integration now face? (RQ1) What suggestions have been made in the literature to address these issues? (RQ2) What are the following issues and research directions for integrating the BIoT? (RQ3). This paper contributes by offering a thorough analysis and synthesis of all BIoT problems found in the literature, as stated in Section 2.4. This paper offers a comprehensive assessment of all potential issues from all known BIoT applications, in contrast to earlier survey studies that only revealed select BIoT challenges based on specific BIoT applications. There were seven categories of challenges totaling 28 challenges. the current best practices were also outlined. Future issues of BIoT integration were also identified and discussed.

Big data analysis, AI, quantum computing, double-chained IoT security scheme, interoperability of IoT and BC, and integration of IoT, BC, and edge computing were the seven future problems mentioned in this article. Future issues will take on a new dimension as a result of these tendencies. For instance, even though many authors mentioned integrating the IoT with other layers such as edge or fog computing, these technologies also have their own issues that should be taken into account when implementing the IoT, such as scalability, interoperability, and the absence of industry standards among different providers. Quantum computing will also pose a serious threat to all already employed security solutions, including BC technology. Therefore, while adapting these countermeasures to the IoT age, future research and industry ideas should take this into account.

According to an examination of related literature, it was determined that BIoT integration may take many different shapes and designs, depending on the desired goal, application, and difficulties. In addition, the studies that were examined offered a range of approaches to address some of these issues. Others focused on minimizing specific problems, while others focused on the overall architectural perspective necessary for the integration. As a result, there is a growing requirement for an efficient design that considers the integration process's obstacles, such as IoT device limitations, privacy, security, and big data analytics. In addition, the best approach for facilitating proper smart contract implementation should be studied. In addition, several other academics have used the integration as a platform for deploying specific applications. Many issues, however, still require additional investigation and answers. Since BC technology is recent and still evolving, data are scarce (e.g., out of date or unavailable due to the rapid and continuous development process), and IoT implementations are still in the early stages. As a result, the data and knowledge we were able to collect from online accessible databases for each platform limited our comprehensive comparison. We concentrated on the most common platforms. Lastly, we did not conduct any real-world tests to ensure that the reported transaction speed was comparable to that stated in the reported studies.

6. Conclusions

As the age of information technology draws to an end, the emerging BC technology and knowledge automation are expected to dominate smart technology in the future. The technological properties of BCs, as well as their development potential, have long been recognized for their far-reaching influence on the actual world. Since Bitcoin's rise to prominence, the BC has quickly progressed, which will change the IoT environment and help other technologies and sectors. The state of the art of BIoT-related problems was evaluated in this article. We went through the history of BC, as well as its characteristics, structure, and platforms. Following that, we presented existing BioT integration difficulties, as well as suggested solutions.

Despite ongoing efforts to develop a successful BIoT integration, several challenges restrict appropriate implementation, as well as the integrated system's application range to ensure its optimal use. Security, privacy preservation, technological concerns, scalability, computational processing, regulations, and BIoT design are the seven categories of challenges we found. Future BIoT problems and research directions were also discussed. Lastly, future BIoT integration trends and developments were explored. By providing more information on the present and potential future barriers that should be taken into account in BIoT integration, the findings of this study may be helpful to practitioners and researchers in the field of BIoT.

Author Contributions: Conceptualization, Y.I.A. and A.A.-A.; methodology, Y.I.A. and H.K.; validation, A.J., A.A.-A. and H.K.; formal analysis, Y.I.A.; writing—original draft preparation, Y.I.A.; writing—review and editing, A.J., A.A.-A. and H.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable, the study does not report any data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Elbasi, E.; Topcu, A.E.; Mathew, S. Prediction of COVID-19 risk in public areas using IoT and machine learning. *Electronics* **2021**, *10*, 1677. [CrossRef]
2. Thakur, N.; Han, C.Y. Indoor localization for personalized ambient assisted living of multiple users in multi-floor smart environments. *Big Data Cogn. Comput.* **2021**, *5*, 42. [CrossRef]
3. Alzoubi, Y.I.; Osmanaj, V.H.; Jaradat, A.; Al-Ahmad, A. Fog computing security and privacy for the internet of thing applications: State-of-the-art. *Secur. Priv.* **2021**, *4*, e145. [CrossRef]

4. Fernández-Caramés, T.M.; Fraga-Lamas, P. A review on the use of blockchain for the internet of things. *IEEE Access* **2018**, *6*, 32979–33001. [CrossRef]
5. Ismail, S.; Almayouf, R.; Chehab, S.; Alghamdi, S.; Almutairi, A.; Alasmari, B.; Altherwy, R. Edge IoT-cloud framework based on blockchain. In Proceedings of the 2020 2nd International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 13–15 October 2020; pp. 1–7.
6. Powell, W.; Foth, M.; Cao, S.; Natanelov, V. Garbage in garbage out: The precarious link between IoT and blockchain in food supply chains. *J. Ind. Inf. Integr.* **2022**, *25*, 100261. [CrossRef]
7. Al-Ahmad, A.S.; Kahtan, H. Cloud computing review: Features and issues. In Proceedings of the 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE), Shah Alam, Malaysia, 11–12 July 2018; pp. 1–5.
8. Abdelmaboud, A.; Ahmed, A.I.A.; Abaker, M.; Eisa, T.A.E.; Albasheer, H.; Ghorashi, S.A.; Karim, F.K. Blockchain for IoT Applications: Taxonomy, Platforms, Recent Advances, Challenges and Future Research Directions. *Electronics* **2022**, *11*, 630. [CrossRef]
9. Bala, K.; Kaur, P.D. Changing trends of blockchain in IoT: Benefits and challenges. In Proceedings of the 2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 27–28 January 2022; pp. 324–329.
10. Brotsis, S.; Limniotis, K.; Bendiab, G.; Kolokotronis, N.; Shiaeles, S. On the suitability of blockchain platforms for IoT applications: Architectures, security, privacy, and performance. *Comput. Netw.* **2021**, *191*, 108005. [CrossRef]
11. Al Sadawi, A.; Hassan, M.S.; Ndiaye, M. A survey on the integration of blockchain with IoT to enhance performance and eliminate challenges. *IEEE Access* **2021**, *9*, 54478–54497. [CrossRef]
12. Hu, S.; Huang, S.; Huang, J.; Su, J. Blockchain and edge computing technology enabling organic agricultural supply chain: A framework solution to trust crisis. *Comput. Ind. Eng.* **2021**, *153*, 107079. [CrossRef]
13. Bhushan, B.; Khamparia, A.; Sagayam, K.M.; Sharma, S.K.; Ahad, M.A.; Debnath, N.C. Blockchain for smart cities: A review of architectures, integration trends and future research directions. *Sustain. Cities Soc.* **2020**, *61*, 102360. [CrossRef]
14. Nakamoto, S.; Bitcoin, A. A peer-to-peer electronic cash system. *Bitcoin* **2008**, *4*, 2.
15. Li, X.; Lu, W.; Xue, F.; Wu, L.; Zhao, R.; Lou, J.; Xu, J. Blockchain-Enabled IoT-BIM Platform for Supply Chain Management in Modular Construction. *J. Constr. Eng. Manag.* **2022**, *148*, 04021195. [CrossRef]
16. Rayes, A.; Salam, S. The Blockchain in IoT. In *Internet of Things from Hype to Reality*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 277–303.
17. Alzoubi, Y.I.; Al-Ahmad, A.; Jaradat, A. Fog computing security and privacy issues, open challenges, and blockchain solution: An overview. *Int. J. Electr. Comput. Eng.* **2021**, *11*, 5081–5088. [CrossRef]
18. Khan, N.S.; Chishti, M.A. Security challenges in fog and IoT, blockchain technology and cell tree solutions: A review. *Scalable Comput.* **2020**, *21*, 515–542. [CrossRef]
19. Aloqaily, M.; Bouachir, O.; Boukerche, A.; Al Ridhawi, I. Design guidelines for blockchain-assisted 5g-uav networks. *IEEE Netw.* **2021**, *35*, 64–71. [CrossRef]
20. Alzoubi, Y.I.; Al-Ahmad, A.; Jaradat, A.; Osmanaj, V.H. Fog computing architecture, benefits, security, and privacy, for the internet of thing applications: An overview. *J. Theor. Appl. Inf. Technol.* **2021**, *99*, 436–451.
21. Qatawneh, M.; Almobaideen, W.; AbuAlghanam, O. Challenges of blockchain technology in context internet of things: A survey. *Int. J. Comput. Appl.* **2020**, *175*, 14–20. [CrossRef]
22. Baouya, A.; Chehida, S.; Bensalem, S.; Bozga, M. Fog computing and blockchain for massive IoT deployment. In Proceedings of the 9th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 8–11 June 2020; pp. 1–4.
23. Srivastava, A.; Dashora, K. Application of blockchain technology for agrifood supply chain management: A systematic literature review on benefits and challenges. *Benchmarking Int. J.* **2022**. [CrossRef]
24. Arslan, S.S.; Jurdak, R.; Jelitto, J.; Krishnamachari, B. Advancements in distributed ledger technology for internet of things. *Internet Things* **2020**, *9*, 100114. [CrossRef]
25. Ali, M.S.; Vecchio, M.; Pincheira, M.; Dolui, K.; Antonelli, F.; Rehmani, M.H. Applications of blockchains in the internet of things: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 1676–1717. [CrossRef]
26. Xie, J.; Tang, H.; Huang, T.; Yu, F.R.; Xie, R.; Liu, J.; Liu, Y. A survey of blockchain technology applied to smart cities: Research issues and challenges. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2794–2830. [CrossRef]
27. Zafar, S.; Bhatti, K.; Shabbir, M.; Hashmat, F.; Akbar, A. Integration of blockchain and Internet of Things: Challenges and solutions. *Ann. Telecommun.* **2022**, *77*, 13–32. [CrossRef]
28. Tsang, Y.; Wu, C.; Ip, W.; Shiau, W.-L. Exploring the intellectual cores of the blockchain–Internet of Things (BIoT). *J. Enterp. Inf. Manag.* **2021**, *34*, 1287–1317. [CrossRef]
29. Huang, J.; Kong, L.; Chen, G.; Wu, M.-Y.; Liu, X.; Zeng, P. Towards secure industrial IoT: Blockchain system with credit-based consensus mechanism. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3680–3689. [CrossRef]
30. Sharma, P.K.; Park, J.H. Blockchain based hybrid network architecture for the smart city. *Future Gener. Comput. Syst.* **2018**, *86*, 650–655. [CrossRef]
31. Ouaddah, A.; Abou Elkalam, A.; Ouahman, A.A. Towards a novel privacy-preserving access control model based on blockchain technology in IoT. In *Europe and MENA Cooperation Advances in Information and Communication Technologies*; Springer: Cham, Switzerland, 2017; Volume 520, pp. 523–533.

32. Alzubi, J.A. Blockchain-based Lamport Merkle Digital Signature: Authentication tool in IoT healthcare. *Comput. Commun.* **2021**, *170*, 200–208. [CrossRef]
33. Rahulamathavan, Y.; Phan, R.C.-W.; Rajarajan, M.; Misra, S.; Kondoz, A. Privacy-preserving blockchain based IoT ecosystem using attribute-based encryption. In Proceedings of the 2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), Bhubaneswar, India, 17–20 December 2017; pp. 1–6.
34. Lu, Y. Blockchain and the related issues: A review of current research topics. *J. Manag. Anal.* **2018**, *5*, 231–255. [CrossRef]
35. Jo, B.W.; Khan, R.M.A.; Lee, Y.-S. Hybrid blockchain and internet-of-things network for underground structure health monitoring. *Sensors* **2018**, *18*, 4268. [CrossRef]
36. Yang, R.; Yu, F.R.; Si, P.; Yang, Z.; Zhang, Y. Integrated blockchain and edge computing systems: A survey, some research issues and challenges. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 1508–1532. [CrossRef]
37. Abdellatif, A.A.; Samara, L.; Mohamed, A.; Erbad, A.; Chiasserini, C.F.; Guizani, M.; O'Connor, M.D.; Laughton, J. MEdge-Chain: Leveraging edge computing and blockchain for efficient medical data exchange. *IEEE Internet Things J.* **2021**, *8*, 15762–15775. [CrossRef]
38. Berdik, D.; Otoum, S.; Schmidt, N.; Porter, D.; Jararweh, Y. A survey on blockchain for information systems management and security. *Inf. Process. Manag.* **2021**, *58*, 102397. [CrossRef]
39. Chang, Z.; Guo, W.; Guo, X.; Chen, T.; Min, G.; Abualnaja, K.M.; Mumtaz, S. Blockchain-Empowered drone networks: Architecture, features, and future. *IEEE Netw.* **2021**, *35*, 86–93. [CrossRef]
40. Yuan, P.; Zheng, K.; Xiong, X.; Zhang, K.; Lei, L. Performance modeling and analysis of a hyperledger-based system using GSPN. *Comput. Commun.* **2020**, *153*, 117–124. [CrossRef]
41. Yang, H.-K.; Cha, H.-J.; Song, Y.-J. Secure identifier management based on blockchain technology in NDN environment. *IEEE Access* **2018**, *7*, 6262–6268. [CrossRef]
42. Rizzardi, A.; Sicari, S.; Miorandi, D.; Coen-Porisini, A. Securing the access control policies to the Internet of Things resources through permissioned blockchain. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e6934. [CrossRef]
43. Kuo, T.-T.; Zavaleta Rojas, H.; Ohno-Machado, L. Comparison of blockchain platforms: A systematic review and healthcare examples. *J. Am. Med. Inform. Assoc.* **2019**, *26*, 462–478. [CrossRef]
44. Paulavičius, R.; Grigaitis, S.; Igumenov, A.; Filatovas, E. A decade of blockchain: Review of the current status, challenges, and future directions. *Informatica* **2019**, *30*, 729–748. [CrossRef]
45. Lone, A.H.; Naaz, R. Applicability of Blockchain smart contracts in securing Internet and IoT: A systematic literature review. *Comput. Sci. Rev.* **2021**, *39*, 100360. [CrossRef]
46. Reyna, A.; Martín, C.; Chen, J.; Soler, E.; Díaz, M. On blockchain and its integration with IoT. Challenges and opportunities. *Future Gener. Comput. Syst.* **2018**, *88*, 173–190. [CrossRef]
47. Wang, S.; Ouyang, L.; Yuan, Y.; Ni, X.; Han, X.; Wang, F.-Y. Blockchain-enabled smart contracts: Architecture, applications, and future trends. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2266–2277. [CrossRef]
48. Griggs, K.N.; Ossipova, O.; Kohlios, C.P.; Baccarini, A.N.; Howson, E.A.; Hayajneh, T. Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. *J. Med. Syst.* **2018**, *42*, 130. [CrossRef] [PubMed]
49. Ghandour, A.G.; Elhoseny, M.; Hassanien, A.E. Blockchains for smart cities: A survey. In *Security in Smart Cities: Models, Applications, and Challenges*; Hassanien, A.E., Elhoseny, M., Ahmed, S., Singh, A., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 193–210.
50. Wang, X.; Zha, X.; Ni, W.; Liu, R.P.; Guo, Y.J.; Niu, X.; Zheng, K. Survey on blockchain for internet of things. *Comput. Commun.* **2019**, *136*, 10–29. [CrossRef]
51. Fotiou, N.; Siris, V.A.; Polyzos, G.C. Interacting with the Internet of Things Using Smart Contracts and Blockchain Technologies. In *Security, Privacy, and Anonymity in Computation, Communication, and Storage: SpaCCS 2018*; Wang, G., Chen, J., Yang, L., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2018. [CrossRef]
52. Mercan, S.; Kurt, A.; Akkaya, K.; Erdin, E. Cryptocurrency solutions to enable micropayments in consumer IoT. *IEEE Consum. Electron. Mag.* **2021**, *11*, 97–103. [CrossRef]
53. Pennino, D.; Pizzonia, M.; Vitaletti, A.; Zecchini, M. Blockchain as IoT Economy enabler: A review of architectural aspects. *J. Sens. Actuator Netw.* **2022**, *11*, 20. [CrossRef]
54. Klein, M.; Stummer, C. Feeless micropayments as drivers for new business models: Two exemplary application cases. *Front. Blockchain* **2021**, *4*, 641508. [CrossRef]
55. Pajooh, H.H.; Rashid, M.; Alam, F.; Demidenko, S. Hyperledger fabric blockchain for securing the edge internet of things. *Sensors* **2021**, *21*, 359. [CrossRef]
56. Pincheira, M.; Antonini, M.; Vecchio, M. Integrating the IoT and Blockchain Technology for the Next Generation of Mining Inspection Systems. *Sensors* **2022**, *22*, 899. [CrossRef]
57. Majeed, U.; Khan, L.U.; Yaqoob, I.; Kazmi, S.A.; Salah, K.; Hong, C.S. Blockchain for IoT-based smart cities: Recent advances, requirements, and future challenges. *J. Netw. Comput. Appl.* **2021**, *181*, 103007. [CrossRef]
58. Uddin, M.A.; Stranieri, A.; Gondal, I.; Balasubramanian, V. A survey on the adoption of blockchain in IoT: Challenges and solutions. *Blockchain Res. Appl.* **2021**, *2*, 100006. [CrossRef]
59. Lashkari, B.; Musilek, P. A comprehensive review of blockchain consensus mechanisms. *IEEE Access* **2021**, *9*, 43620–43652. [CrossRef]

60. Turk, Ž.; Klinc, R. Potentials of blockchain technology for construction management. *Procedia Eng.* **2017**, *196*, 638–645. [CrossRef]
61. Kumar, N.M.; Mallick, P.K. Blockchain technology for security issues and challenges in IoT. *Procedia Comput. Sci.* **2018**, *132*, 1815–1823. [CrossRef]
62. Pahl, C.; El Ioini, N.; Helmer, S. A decision framework for blockchain platforms for IoT and edge computing. In Proceedings of the IoTBDS 2018, Madeira, Purtogal, 19–21 March 2018. [CrossRef]
63. Ferrag, M.A.; Derdour, M.; Mukherjee, M.; Derhab, A.; Maglaras, L.; Janicke, H. Blockchain technologies for the internet of things: Research issues and challenges. *IEEE Internet Things J.* **2018**, *6*, 2188–2204. [CrossRef]
64. Alladi, T.; Chamola, V.; Parizi, R.M.; Choo, K.-K.R. Blockchain applications for industry 4.0 and industrial IoT: A review. *IEEE Access* **2019**, *7*, 176935–176951. [CrossRef]
65. Lee, J.; Azamfar, M.; Singh, J. A blockchain enabled cyber-physical system architecture for industry 4.0 manufacturing systems. *Manuf. Lett.* **2019**, *20*, 34–39. [CrossRef]
66. Wei, L.; Wu, J.; Long, C.; Lin, Y.-B. The convergence of ioe and blockchain: Security challenges. *IT Prof.* **2019**, *21*, 26–32. [CrossRef]
67. Ahmed, S.; Shah, M.A.; Wakil, K. Blockchain as a trust builder in the smart city domain: A systematic literature review. *IEEE Access* **2020**, *8*, 92977–92985. [CrossRef]
68. Ferrag, M.A.; Shu, L.; Yang, X.; Derhab, A.; Maglaras, L. Security and privacy for green IoT-based agriculture: Review, blockchain solutions, and challenges. *IEEE Access* **2020**, *8*, 32031–32053. [CrossRef]
69. Rao, A.R.; Clarke, D. Perspectives on emerging directions in using IoT devices in blockchain applications. *Internet Things* **2020**, *10*, 100079. [CrossRef]
70. Wang, Q.; Zhu, X.; Ni, Y.; Gu, L.; Zhu, H. Blockchain for the IoT and industrial IoT: A review. *Internet Things* **2020**, *10*, 100081. [CrossRef]
71. Tseng, L.; Yao, X.; Otoum, S.; Aloqaily, M.; Jararweh, Y. Blockchain-based database in an IoT environment: Challenges, opportunities, and analysis. *Clust. Comput.* **2020**, *23*, 2151–2165. [CrossRef]
72. Garay, J.; Kiayias, A.; Leonardos, N. The bitcoin backbone protocol: Analysis and applications. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, 26–30 April 2015; pp. 281–310.
73. Bhushan, B.; Sahoo, C.; Sinha, P.; Khamparia, A. Unification of blockchain and internet of things (BIoT): Requirements, working model, challenges and future directions. *Wirel. Netw.* **2021**, *27*, 55–90. [CrossRef]
74. Farahani, B.; Firouzi, F.; Luecking, M. The convergence of IoT and distributed ledger technologies (DLT): Opportunities, challenges, and solutions. *J. Netw. Comput. Appl.* **2021**, *177*, 102936. [CrossRef]
75. Singh, S.; Hosen, A.S.; Yoon, B. Blockchain security attacks, challenges, and solutions for the future distributed IoT network. *IEEE Access* **2021**, *9*, 13938–13959. [CrossRef]
76. Da Xu, L.; Lu, Y.; Li, L. Embedding blockchain technology into IoT for security: A survey. *IEEE Internet Things J.* **2021**, *8*, 10452–10473.
77. Yaqoob, I.; Salah, K.; Jayaraman, R.; Al-Hammadi, Y. Blockchain for healthcare data management: Opportunities, challenges, and future recommendations. *Neural Comput. Appl.* **2022**, *34*, 11475–11490. [CrossRef]
78. Kumar, R.L.; Khan, F.; Kadry, S.; Rho, S. A Survey on blockchain for industrial Internet of Things. *Alex. Eng. J.* **2022**, *61*, 6001–6022. [CrossRef]
79. Yu, Z.; Song, L.; Jiang, L.; Sharafi, O.K. Systematic literature review on the security challenges of blockchain in IoT-based smart cities. *Kybernetes* **2021**, *51*. [CrossRef]
80. Alkhateeb, A.; Catal, C.; Kar, G.; Mishra, A. Hybrid blockchain platforms for the internet of things (IoT): A systematic literature review. *Sensors* **2022**, *22*, 1304. [CrossRef]
81. Holst, A. Number of IoT Connected Devices Worldwide 2019–2030. *Statistica* **2022**. Available online: https://www.statista.com/statistics/1183463/iot-connected-devices-worldwide-by-technology/ (accessed on 29 June 2022).
82. Gill, S.S. Quantum and blockchain based Serverless edge computing: A vision, model, new trends and future directions. *Internet Technol. Lett.* **2021**, e275. [CrossRef]
83. Dorri, A.; Kanhere, S.S.; Jurdak, R.; Gauravaram, P. Blockchain for IoT security and privacy: The case study of a smart home. In Proceedings of the 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom workshops), Kona, HI, USA, 13 March 2017; pp. 618–623.
84. Esposito, C.; Ficco, M.; Gupta, B.B. Blockchain-based authentication and authorization for smart city applications. *Inf. Process. Manag.* **2021**, *58*, 102468. [CrossRef]
85. Khan, M.A.; Salah, K. IoT security: Review, blockchain solutions, and open challenges. *Future Gener. Comput. Syst.* **2018**, *82*, 395–411. [CrossRef]
86. Zhang, W.; Wu, Z.; Han, G.; Feng, Y.; Shu, L. Ldc: A lightweight dada consensus algorithm based on the blockchain for the industrial internet of things for smart city applications. *Future Gener. Comput. Syst.* **2020**, *108*, 574–582. [CrossRef]
87. Zhong, L.; Wu, Q.; Xie, J.; Guan, Z.; Qin, B. A secure large-scale instant payment system based on blockchain. *Comput. Secur.* **2019**, *84*, 349–364. [CrossRef]
88. Li, X.; Jiang, P.; Chen, T.; Luo, X.; Wen, Q. A survey on the security of blockchain systems. *Future Gener. Comput. Syst.* **2020**, *107*, 841–853. [CrossRef]

89. Abdi, A.I.; Eassa, F.E.; Jambi, K.; Almarhabi, K.; Khemakhem, M.; Basuhail, A.; Yamin, M. Hierarchical Blockchain-Based Multi-Chaincode Access Control for Securing IoT Systems. *Electronics* **2022**, *11*, 711. [CrossRef]
90. Alzahrani, N.; Bulusu, N. Towards true decentralization: A blockchain consensus protocol based on game theory and randomness. In Proceedings of the International Conference on Decision and Game Theory for Security, Cham, Switzerland, 29–31 October 2018; pp. 465–485.
91. Liao, C.-F.; Bao, S.-W.; Cheng, C.-J.; Chen, K. On design issues and architectural styles for blockchain-driven IoT services. In Proceedings of the 2017 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), Taipei, Taiwan, 12–14 June 2017; pp. 351–352.
92. Kim, H.; Park, J.; Bennis, M.; Kim, S.-L. Blockchained on-device federated learning. *IEEE Commun. Lett.* **2019**, *24*, 1279–1283. [CrossRef]
93. Vivar, A.L.; Orozco, A.L.S.; Villalba, L.J.G. A security framework for ethereum smart contracts. *Comput. Commun.* **2021**, *172*, 119–129. [CrossRef]
94. Abbasi, Y.; Benlahmer, H. BCSDN-IoT: Towards an IoT security architecture based on SDN and Blockchain. *Int. J. Electr. Comput. Eng. Syst.* **2022**, *13*, 155–163. [CrossRef]
95. Latif, S.A.; Wen, F.B.X.; Iwendi, C.; Li-li, F.W.; Mohsin, S.M.; Han, Z.; Band, S.S. AI-empowered, blockchain and SDN integrated security architecture for IoT network of cyber physical systems. *Comput. Commun.* **2022**, *181*, 274–283. [CrossRef]
96. Qiu, J.; Liang, X.; Shetty, S.; Bowden, D. Towards secure and smart healthcare in smart cities using blockchain. In Proceedings of the 2018 IEEE International Smart Cities Conference (ISC2), Kansas City, MO, USA, 16–19 September 2018; pp. 1–4.
97. Lakhan, A.; Mohammed, M.A.; Kadry, S.; AlQahtani, S.A.; Maashi, M.S.; Abdulkareem, K.H. Federated Learning-Aware Multi-Objective Modeling and blockchain-enable system for IIoT applications. *Comput. Electr. Eng.* **2022**, *100*, 107839. [CrossRef]
98. Hannah, S.; Deepa, A.; Chooralil, V.S.; BrillySangeetha, S.; Yuvaraj, N.; Arshath Raja, R.; Suresh, C.; Vignesh, R.; Srihari, K.; Alene, A. Blockchain-based deep learning to process IoT data acquisition in cognitive data. *BioMed Res. Int.* **2022**, *2022*, 5038851. [CrossRef] [PubMed]
99. Khan, L.U.; Saad, W.; Han, Z.; Hossain, E.; Hong, C.S. Federated learning for internet of things: Recent advances, taxonomy, and open challenges. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1759–1799. [CrossRef]
100. Ghazal, T.M.; Hasan, M.K.; Alshurideh, M.T.; Alzoubi, H.M.; Ahmad, M.; Akbar, S.S.; Al Kurdi, B.; Akour, I.A. IoT for smart cities: Machine learning approaches in smart healthcare—A review. *Future Internet* **2021**, *13*, 218. [CrossRef]
101. Bouras, M.; Lu, Q.; Dhelim, S.; Ning, H. A Lightweight Blockchain-Based IoT Identity Management Approach. *Future Internet* **2021**, *13*, 24. [CrossRef]
102. Du, Y.; Wang, Z.; Leung, V. Blockchain-Enabled edge intelligence for IoT: Background, emerging trends and open issues. *Future Internet* **2021**, *13*, 48. [CrossRef]
103. Ang, K.L.M.; Seng, J.K.P.; Ngharamike, E. Towards crowdsourcing internet of things (crowd-iot): Architectures, security and applications. *Future Internet* **2022**, *14*, 49. [CrossRef]
104. Tomer, V.; Sharma, S. Detecting IoT Attacks Using an Ensemble Machine Learning Model. *Future Internet* **2022**, *14*, 102. [CrossRef]
105. Yazdinejad, A.; Parizi, R.M.; Dehghantanha, A.; Zhang, Q.; Choo, K.-K.R. An energy-efficient SDN controller architecture for IoT networks with blockchain-based security. *IEEE Trans. Serv. Comput.* **2020**, *13*, 625–638. [CrossRef]
106. Li, C.; Zhang, L.-J. A blockchain based new secure multi-layer network model for internet of things. In Proceedings of the 2017 IEEE International Congress on Internet of Things (ICIOT), Honolulu, HI, USA, 25–30 June 2017; pp. 33–41.
107. Hasankhani, A.; Hakimi, S.M.; Shafie-khah, M.; Asadolahi, H. Blockchain technology in the future smart grids: A comprehensive review and frameworks. *Int. J. Electr. Power Energy Syst.* **2021**, *129*, 106811. [CrossRef]
108. Christidis, K.; Devetsikiotis, M. Blockchains and smart contracts for the internet of things. *IEEE Access* **2016**, *4*, 2292–2303. [CrossRef]
109. Chen, J.; Duan, K.; Zhang, R.; Zeng, L.; Wang, W. An AI based super nodes selection algorithm in blockchain networks. *arXiv* **2018**, arXiv:1808.00216.
110. Chen, F.; Xiao, Z.; Cui, L.; Lin, Q.; Li, J.; Yu, S. Blockchain for internet of things applications: A review and open issues. *J. Netw. Comput. Appl.* **2020**, *172*, 102839. [CrossRef]
111. Wang, J.; Liu, Y.; Niu, S.; Song, H. Lightweight blockchain assisted secure routing of swarm UAS networking. *Comput. Commun.* **2021**, *165*, 131–140. [CrossRef]
112. Hakak, S.; Khan, W.Z.; Gilkar, G.A.; Imran, M.; Guizani, N. Securing smart cities through blockchain technology: Architecture, requirements, and challenges. *IEEE Netw.* **2020**, *34*, 8–14. [CrossRef]
113. Suhail, S.; Hussain, R.; Jurdak, R.; Hong, C.S. Trustworthy digital twins in the industrial internet of things with blockchain. *IEEE Internet Comput.* **2021**, *26*, 58–67. [CrossRef]
114. Sengupta, J.; Ruj, S.; Bit, S.D. A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT. *J. Netw. Comput. Appl.* **2020**, *149*, 102481. [CrossRef]
115. Shen, M.; Tang, X.; Zhu, L.; Du, X.; Guizani, M. Privacy-preserving support vector machine training over blockchain-based encrypted IoT data in smart cities. *IEEE Internet Things J.* **2019**, *6*, 7702–7712. [CrossRef]
116. Singh, S.; Sharma, P.K.; Yoon, B.; Shojafar, M.; Cho, G.H.; Ra, I.-H. Convergence of blockchain and artificial intelligence in IoT network for the sustainable smart city. *Sustain. Cities Soc.* **2020**, *63*, 102364. [CrossRef]

117. Tan, L.; Shi, N.; Yang, C.; Yu, K. A blockchain-based access control framework for cyber-physical-social system big data. *IEEE Access* **2020**, *8*, 77215–77226. [CrossRef]
118. Uddin, M.A.; Stranieri, A.; Gondal, I.; Balasubramanian, V. Blockchain leveraged decentralized iot ehealth framework. *Internet Things* **2020**, *9*, 100159. [CrossRef]
119. Vivekanandan, M.; Sastry, V. BIDAPSCA5G: Blockchain based internet of things (IoT) device to device authentication protocol for smart city applications using 5G technology. *Peer-to-Peer Netw. Appl.* **2021**, *14*, 403–419. [CrossRef]
120. Anitha, A.; Haritha, T. The Integration of Blockchain With IoT in Smart Appliances: A Systematic Review. In *Blockchain Technologies for Sustainable Development in Smart Cities*; IGI Global: Hershey, PA, USA, 2022; pp. 223–246.
121. Arul, R.; Al-Otaibi, Y.D.; Alnumay, W.S.; Tariq, U.; Shoaib, U.; Piran, M.J. Multi-modal secure healthcare data dissemination framework using blockchain in IoMT. *Pers. Ubiquitous Comput.* **2021**. [CrossRef]
122. Awan, S.H.; Ahmed, S.; Nawaz, A.; Sulaiman, S.; Zaman, K.; Ali, M.; Najam, Z.; Imran, S. BlockChain with IoT, an emergent routing scheme for smart agriculture. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 420–429. [CrossRef]
123. Banerjee, S.; Bera, B.; Das, A.K.; Chattopadhyay, S.; Khan, M.K.; Rodrigues, J.J. Private blockchain-envisioned multi-authority CP-ABE-based user access control scheme in IIoT. *Comput. Commun.* **2021**, *169*, 99–113. [CrossRef]
124. Bera, B.; Chattaraj, D.; Das, A.K. Designing secure blockchain-based access control scheme in IoT-enabled Internet of Drones deployment. *Comput. Commun.* **2020**, *153*, 229–249. [CrossRef]
125. Bhawiyuga, A.; Wardhana, A.; Amron, K.; Kirana, A.P. Platform for integrating internet of things based smart healthcare system and blockchain network. In Proceedings of the 6th NAFOSTED Conference on Information and Computer Science (NICS), Hanoi, Vietnam, 12–13 December 2019; pp. 55–60.
126. Brandão, A.; São Mamede, H.; Gonçalves, R. Systematic review of the literature, research on blockchain technology as support to the trust model proposed applied to smart places. In Proceedings of the World Conference on Information Systems and Technologies, Budva, Montenegro, 27–29 March 2018; pp. 1163–1174.
127. Dagher, G.G.; Mohler, J.; Milojkovic, M.; Marella, P.B. Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology. *Sustain. Cities Soc.* **2018**, *39*, 283–297. [CrossRef]
128. Devi, M.S.; Suguna, R.; Joshi, A.S.; Bagate, R.A. Design of IoT blockchain based smart agriculture for enlightening safety and security. In Proceedings of the International Conference on Emerging Technologies in Computer Engineering, Jaipur, India, 1–2 February 2019; pp. 7–19.
129. Dwivedi, A.D.; Srivastava, G.; Dhar, S.; Singh, R. A decentralized privacy-preserving healthcare blockchain for IoT. *Sensors* **2019**, *19*, 326. [CrossRef]
130. El Kafhali, S.; Chahir, C.; Hanini, M.; Salah, K. Architecture to manage internet of things data using blockchain and fog computing. In Proceedings of the 4th International Conference on Big Data and Internet of Things, Rabat, Morocco, 23–24 October 2019; pp. 1–8.
131. Farouk, A.; Alahmadi, A.; Ghose, S.; Mashatan, A. Blockchain platform for industrial healthcare: Vision and future opportunities. *Comput. Commun.* **2020**, *154*, 223–235. [CrossRef]
132. Hewa, T.; Ylianttila, M.; Liyanage, M. Survey on blockchain based smart contracts: Applications, opportunities and challenges. *J. Netw. Comput. Appl.* **2020**, *177*, 102857. [CrossRef]
133. Huang, J.-C.; Shu, M.-H.; Hsu, B.-M.; Hu, C.-M. Service architecture of IoT terminal connection based on blockchain identity authentication system. *Comput. Commun.* **2020**, *160*, 411–422. [CrossRef]
134. Huh, S.; Cho, S.; Kim, S. Managing IoT devices using blockchain platform. In Proceedings of the 19th International Conference on Advanced Communication Technology (ICACT), PyeongChang, Korea, 19–22 February 2017; pp. 464–467.
135. Khan, F.A.; Asif, M.; Ahmad, A.; Alharbi, M.; Aljuaid, H. Blockchain technology, improvement suggestions, security challenges on smart grid and its application in healthcare for sustainable development. *Sustain. Cities Soc.* **2020**, *55*, 102018. [CrossRef]
136. Kumari, A.; Gupta, R.; Tanwar, S.; Kumar, N. A taxonomy of blockchain-enabled softwarization for secure UAV network. *Comput. Commun.* **2020**, *161*, 304–323. [CrossRef]
137. McGhin, T.; Choo, K.-K.R.; Liu, C.Z.; He, D. Blockchain in healthcare applications: Research challenges and opportunities. *J. Netw. Comput. Appl.* **2019**, *135*, 62–75. [CrossRef]
138. Mehta, P.; Gupta, R.; Tanwar, S. Blockchain envisioned UAV networks: Challenges, solutions, and comparisons. *Comput. Commun.* **2020**, *151*, 518–538. [CrossRef]
139. Memon, R.A.; Li, J.P.; Nazeer, M.I.; Khan, A.N.; Ahmed, J. DualFog-IoT: Additional fog layer for solving blockchain integration problem in internet of things. *IEEE Access* **2019**, *7*, 169073–169093. [CrossRef]
140. Miglani, A.; Kumar, N.; Chamola, V.; Zeadally, S. Blockchain for internet of energy management: Review, solutions, and challenges. *Comput. Commun.* **2020**, *151*, 395–418. [CrossRef]
141. Minoli, D.; Occhiogrosso, B. Blockchain mechanisms for IoT security. *Internet Things* **2018**, *1*, 1–13. [CrossRef]
142. Misra, S.; Deb, P.K.; Pathak, N.; Mukherjee, A. Blockchain-enabled SDN for securing fog-based resource-constrained IoT. In Proceedings of the INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 6–9 July 2020; pp. 490–495.
143. Moin, S.; Karim, A.; Safdar, Z.; Safdar, K.; Ahmed, E.; Imran, M. Securing IoTs in distributed blockchain: Analysis, requirements and open issues. *Future Gener. Comput. Syst.* **2019**, *100*, 325–343. [CrossRef]

144. Naseer, O.; Ullah, S.; Anjum, L. Blockchain-based decentralized lightweight control access scheme for smart grids. *Arab. J. Sci. Eng.* **2021**, *46*, 8233–8243. [CrossRef]
145. Panarello, A.; Tapas, N.; Merlino, G.; Longo, F.; Puliafito, A. Blockchain and iot integration: A systematic survey. *Sensors* **2018**, *18*, 2575. [CrossRef] [PubMed]
146. Pavithran, D.; Al-Karaki, J.N.; Shaalan, K. Edge-based blockchain architecture for event-driven IoT using hierarchical identity based encryption. *Inf. Process. Manag.* **2021**, *58*, 102528. [CrossRef]
147. Qu, C.; Tao, M.; Yuan, R. A hypergraph-based blockchain model and application in internet of things-enabled smart homes. *Sensors* **2018**, *18*, 2784. [CrossRef] [PubMed]
148. Rahman, M.A.; Rashid, M.M.; Hossain, M.S.; Hassanain, E.; Alhamid, M.F.; Guizani, M. Blockchain and IoT-based cognitive edge framework for sharing economy services in a smart city. *IEEE Access* **2019**, *7*, 18611–18621. [CrossRef]
149. Rasool, S.; Iqbal, M.; Dagiuklas, T.; Ul-Qayyum, Z.; Li, S. Reliable data analysis through blockchain based crowdsourcing in mobile ad-hoc cloud. *Mob. Netw. Appl.* **2020**, *25*, 153–163. [CrossRef]
150. Rathore, S.; Kwon, B.W.; Park, J.H. BlockSecIoTNet: Blockchain-based decentralized security architecture for IoT network. *J. Netw. Comput. Appl.* **2019**, *143*, 167–177. [CrossRef]
151. Rifi, N.; Rachkidi, E.; Agoulmine, N.; Taher, N.C. Towards using blockchain technology for IoT data access protection. In Proceedings of the 17th International Conference on Ubiquitous Wireless Broadband (ICUWB), Salamanca, Spain, 12–15 September 2017; pp. 1–5.
152. AlAhmad, A.S.; Kahtan, H.; Alzoubi, Y.I.; Ali, O.; Jaradat, A. Mobile cloud computing models security issues: A systematic review. *J. Netw. Comput. Appl.* **2021**, *190*, 103152. [CrossRef]
153. Ferreira, C.M.S.; Garrocho, C.T.B.; Oliveira, R.A.R.; Silva, J.S.; Cavalcanti, C.F.M. IoT registration and authentication in smart city applications with blockchain. *Sensors* **2021**, *21*, 1323. [CrossRef]
154. Azaria, A.; Ekblaw, A.; Vieira, T.; Lippman, A. Medrec: Using blockchain for medical data access and permission management. In Proceedings of the 2nd International Conference on Open and Big Data (OBD), Vienna, Austria, 22–24 August 2016; pp. 25–30.
155. Mettler, M. Blockchain technology in healthcare: The revolution starts here. In Proceedings of the 18th International Conference on E-health Networking, Applications and Services (Healthcom), Munich, Germany, 14–16 September 2016; pp. 1–3.
156. Srivastava, G.; Parizi, R.M.; Dehghantanha, A. The Future of Blockchain Technology in Healthcare Internet of Things Security. In *Blockchain Cybersecurity, Trust and Privacy*; Choo, K.K., Dehghantanha, A., Parizi, R., Eds.; Advances in Information Security; Springer: Cham, Switzerland, 2020; Volume 79. [CrossRef]
157. Gao, W.; Hatcher, W.G.; Yu, W. A survey of blockchain: Techniques, applications, and challenges. In Proceedings of the 27th International Conference On Computer Communication and Networks (ICCCN), Hangzhou, China, 30 July–2 August 2018; pp. 1–11.
158. Machado, C.; Fröhlich, A.A.M. IoT data integrity verification for cyber-physical systems using blockchain. In Proceedings of the 21st International Symposium on Real-Time Distributed Computing (ISORC), Singapore, 29–31 May 2018; pp. 83–90.
159. Mora, O.B.; Rivera, R.; Larios, V.M.; Beltrán-Ramírez, J.R.; Maciel, R.; Ochoa, A. A use case in cybersecurity based in blockchain to deal with the security and privacy of citizens and smart cities cyberinfrastructures. In Proceedings of the 2018 IEEE International Smart Cities Conference (ISC2), Kansas City, MO, USA, 16–19 September 2018; pp. 1–4.
160. Kiayias, A.; Panagiotakos, G. On trees, chains and fast transactions in the blockchain. In *Progress in Cryptology—LATINCRYPT 2017*; LATINCRYPT 2017. Lecture Notes in Computer Science; Lange, T., Dunkelman, O., Eds.; Springer: Cham, Switzerland, 2017; Volume 11368, pp. 327–351.
161. Natoli, C.; Gramoli, V. The balance attack against proof-of-work blockchains: The R3 testbed as an example. *arXiv* **2016**, arXiv:1612.09426.
162. Alzoubi, Y.I.; Al-Ahmad, A.; Kahtan, H. Blockchain technology as a Fog computing security and privacy solution: An overview. *Comput. Commun.* **2022**, *182*, 129–152. [CrossRef]
163. Ma, M.; Shi, G.; Li, F. Privacy-oriented blockchain-based distributed key management architecture for hierarchical access control in the IoT scenario. *IEEE Access* **2019**, *7*, 34045–34059. [CrossRef]
164. Dang, T.L.N.; Nguyen, M.S. An approach to data privacy in smart home using blockchain technology. In Proceedings of the 2018 International Conference on Advanced Computing and Applications (ACOMP), Ho Chi Minh City, Vietnam, 27–29 November 2018; pp. 58–64.
165. Lin, J.; Shen, Z.; Zhang, A.; Chai, Y. Blockchain and IoT based food traceability for smart agriculture. In Proceedings of the 3rd International Conference on Crowd Science and Engineering, Singapore, 28–31 July 2018; pp. 1–6.
166. Dorri, A.; Kanhere, S.S.; Jurdak, R. Towards an optimized blockchain for IoT. In Proceedings of the 2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI), Pittsburgh, PA, USA, 18–21 April 2017; pp. 173–178.
167. Chaudhry, N.; Yousaf, M.M. Consensus algorithms in blockchain: Comparative analysis, challenges and opportunities. In Proceedings of the 12th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan, 19–21 December 2018; pp. 54–63.
168. Bi, W.; Yang, H.; Zheng, M. An accelerated method for message propagation in blockchain networks. *arXiv* **2018**, arXiv:1809.00455.
169. Ghosh, B.; Bouri, E. Is Bitcoin's carbon footprint persistent? Multifractal evidence and policy implications. *Entropy* **2022**, *24*, 647. [CrossRef]

170. Singh, S.; Ra, I.-H.; Meng, W.; Kaur, M.; Cho, G.H. SH-BlockCC: A secure and efficient internet of things smart home architecture based on cloud computing and blockchain technology. *Int. J. Distrib. Sens. Netw.* **2019**, *15*, 1550147719844159. [CrossRef]
171. Hazari, S.S.; Mahmoud, Q.H. A parallel proof of work to improve transaction speed and scalability in blockchain systems. In Proceedings of the 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; pp. 0916–0921.
172. Simić, M.; Sladić, G.; Milosavljević, B. A case study IoT and blockchain powered healthcare. In Proceedings of the 8th PSU-UNS International Conference on Engineering and Technology (ICET), Novi Sad, Serbia, 13 March 2017.
173. Saghiri, A.M.; Vahdati, M.; Gholizadeh, K.; Meybodi, M.R.; Dehghan, M.; Rashidi, H. A framework for cognitive internet of things based on blockchain. In Proceedings of the 4th International Conference on Web Research (ICWR), Tehran, Iran, 11–12 May 2018; pp. 138–143.
174. Samaniego, M.; Jamsrandorj, U.; Deters, R. Blockchain as a service for IoT. In Proceedings of the 2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Chengdu, China, 15–18 December 2016; pp. 433–436.
175. Stanciu, A. Blockchain based distributed control system for edge computing. In Proceedings of the 21st International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 29–31 May 2017; pp. 667–671.
176. Savazzi, S.; Nicoli, M.; Rampa, V. Federated learning with cooperating devices: A consensus approach for massive IoT networks. *IEEE Internet Things J.* **2020**, *7*, 4641–4654. [CrossRef]
177. Qu, Y.; Gao, L.; Luan, T.H.; Xiang, Y.; Yu, S.; Li, B.; Zheng, G. Decentralized privacy using blockchain-enabled federated learning in fog computing. *IEEE Internet Things J.* **2020**, *7*, 5171–5183. [CrossRef]
178. Outchakoucht, A.; Hamza, E.; Leroy, J.P. Dynamic access control policy based on blockchain and machine learning for the internet of things. *Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 417–424. [CrossRef]
179. Roldán, J.; Boubeta-Puig, J.; Martínez, J.L.; Ortiz, G. Integrating complex event processing and machine learning: An intelligent architecture for detecting IoT security attacks. *Expert Syst. Appl.* **2020**, *149*, 113251. [CrossRef]
180. Ren, Z.; Wu, H.; Ning, Q.; Hussain, I.; Chen, B. End-to-end malware detection for android IoT devices using deep learning. *Ad Hoc Netw.* **2020**, *101*, 102098. [CrossRef]
181. Manogaran, G.; Rawal, B.S.; Saravanan, V.; Kumar, P.M.; Martínez, O.S.; Crespo, R.G.; Montenegro-Marin, C.E.; Krishnamoorthy, S. Blockchain based integrated security measure for reliable service delegation in 6G communication environment. *Comput. Commun.* **2020**, *161*, 248–256. [CrossRef]