Architecting Internet of Things Systems with Blockchain: A Catalog of Tactics

WENDY YÁNEZ, Southern University of Science and Technology, China and University of Birmingham, UK
RAMI BAHSOON, University of Birmingham, UK
YUQUN ZHANG, Southern University of Science and Technology, China
RICK KAZMAN, Software Engineering Institute (SEI)/CMU and University of Hawaii, US

Blockchain offers a distributed ledger to record data collected from Internet of Thing (IoT) devices as immutable and tamper-proof transactions and securely shared among authorized participants in a Peer-to-Peer (P2P) network. Despite the growing interest in using blockchain for securing IoT systems, there is a general lack of systematic research and comprehensive review of the design issues on the integration of blockchain and IoT from the software architecture perspective. This article presents a catalog of architectural tactics for the design of IoT systems supported by blockchain as a result of a Systematic Literature Review (SLR) on IoT and blockchain to extract the commonly reported quality attributes, design decisions, and relevant architectural tactics for the architectural design of this category of systems. Our findings are threefold: (i) identification of security, scalability, performance, and interoperability as the commonly reported quality attributes; (ii) a catalog of twelve architectural tactics for the design of IoT systems supported by blockchain; and (iii) gaps in research that include tradeoffs among quality attributes and identified tactics. These tactics might provide architects and designers with different options when searching for an optimal architectural design that meets the quality attributes of interest and constraints of a system.

CCS Concepts: • **General and reference → Surveys and overviews**; • **Software and its engineering → Requirements analysis**; **Software design tradeoffs**;

Additional Key Words and Phrases: Internet of things, blockchain, software requirements, architectural tactics

ACM Reference format:
Wendy Yánez, Rami Bahsoon, Yuqun Zhang, and Rick Kazman. 2021. Architecting Internet of Things Systems with Blockchain: A Catalog of Tactics. *ACM Trans. Softw. Eng. Methodol.* 30, 3, Article 35 (April 2021), 46 pages.
https://doi.org/10.1145/3442412

This work is partially supported by the National Natural Science Foundation of China (Grant No. 61902169), Shenzhen Peacock Plan (Grant No. KQTD2016112514355531), and Science and Technology Innovation Committee Foundation of Shenzhen (Grant No. JCYJ20170817110848086).

Authors' addresses: W. Yánez (corresponding author), Southern University of Science and Technology, China and University of Birmingham, UK; email: wpy443@student.bham.ac.uk; R. Bahsoon, University of Birmingham, Birmingham, UK; email: r.bahsoon@cs.bham.ac.uk; Y. Zhang (corresponding author), Southern University of Science and Technology, Shenzhen, China; email: zhangyq@sustech.edu.cn; R. Kazman, Software Engineering Institute (SEI)/CMU and University of Hawaii, Hawaii; email: kazman@hawaii.edu.

1049-331X/2021/04-ART35 $15.00
https://doi.org/10.1145/3442412

1 INTRODUCTION

Internet of Things (IoT) comprises a global network of smart devices that collect data from their surroundings and exchange it with each other over the Internet [9, 10, 24, 74, 77] and can be widely applied in multiple domains, e.g., cloud computing [58, 76, 79, 81, 82] and AI-oriented applications [15, 36, 64, 67, 80, 84]. It is predicted that 50 billion of IoT devices will be connected to the Internet by 2020 [19], which will result in the generation of 500 zettabytes of data by 2025 [22]. However, the majority of IoT devices (i.e., sensors and RFID tags) still do not have enough memory, computation, and battery lifetime that will allow them to implement security protocols and make intelligent decisions in healthcare, manufacturing, environmental monitoring, and smart cities [10, 26]. Thus, IoT systems heavily rely on cloud services for data processing and analytics, but such systems could lead to high latency, high maintenance costs, and security and privacy issues [51, 53].

Blockchain is an append-only ledger based on a P2P network and cryptography techniques to provide an immutable and shared data storage, which only allows inserting new transactions without deleting or updating existing ones [70, 72]. Each transactional datum is encrypted using a public-private key pair and broadcast to the P2P network to be validated in consensus before recording it on the ledger [4]. In addition to the shared infrastructure to store data, blockchain facilitates the execution of programs, called smart contracts that codify rules, conditions, and business logic among two or more parties without the need for a trusted party [37]. In particular, a smart contract automatically executes terms of agreements among two parties that have agreed to trade tangible goods or services when the conditions in the agreement are met [1]. There are two prominent blockchain platforms for IoT systems, called Bitcoin and Ethereum, which can record cryptographically signed financial and complex transactions respectively. The Bitcoin protocol identifies two types of blockchain nodes called full and lite nodes that facilitate the adoption of blockchain in IoT. The former has enough processing power and storage capacity to keep a complete copy of the ledger while the latter is not able to keep a complete copy of the blockchain. Instead, they download the block headers to validate the authenticity of the transactions in the blockchain network [42]. With the decentralized nature of blockchain, critical IoT data, and Machine-to-Machine communication can be recorded securely and shared among authorized participants in the P2P network [65]. Moreover, IoT systems can take advantage of smart contracts to configure and manage IoT devices as well as share services and resources among them [20, 52].

Despite the growing interest of academia and the software industry in the integration of blockchain and IoT [45, 53, 68], there is still a need for systematic approaches that support the architectural design of blockchain-enabled IoT systems. To the best of our knowledge, only a few studies [37, 52, 54, 75] manifest the lack of insight into design decisions and tradeoffs among quality attribute to satisfy when architecting this category of systems. Overall, these issues can affect how these systems are conceived, developed, and deployed and could have significant implications on the achievement of quality attributes of interest. Bridging this gap is a prerequisite for realizing the potential of both technologies and facilitating the design of data-centric architectures for blockchain-based IoT systems. In this article, we provide a catalog of architectural tactics for the design of IoT systems supported by blockchain and explain how they can influence the achievement of the quality attributes of interest. To achieve this, we conduct a SLR to investigate the commonly reported quality attributes and design decisions to consider when architecting this category of systems. Next, we extracted the relevant architectural tactics from the reviewed literature to provide architects and designers with transferable architectural knowledge and different options for addressing individual quality attributes. Our approach has followed the guidelines for performing SLR in software engineering as described by Kitchenham and Charters [33].

Our findings are drawn from 100 research papers that were rigorously selected from a repository of 575 peer-reviewed, published articles on blockchain and IoT. In particular, we identify security, scalability, and performance as the commonly reported quality attributes for the design of IoT systems supported by blockchain. In addition, we provide a catalog of 12 architectural tactics that can be used by software architects and designers to achieve the required quality attributes. To describe each tactic, we draw inspiration from the template that covers summary, motivation, description, constraints, examples, and related tactics [35]. It is worth noting that the identified tactics are not exhaustive. Instead, we attempt to provide a categorization of the existing tactics in the literature to guide architects and researchers in the inception and implementation of specialized architectures for IoT systems supported by blockchain. This work reveals that (i) despite the significance of the identified quality attributes, there are other requirements that lack architectural support in the literature; (ii) investigation is required to evaluate the impact of the architectural tactics in this category of systems; and (iii) additional research is needed to explore the tradeoffs among the quality attributes and identified tactics. These opportunities for future research require extensive collaboration between industry and academia to implement, deploy, and evaluate architectural tactics and control quality attributes in large-scale IoT systems supported by blockchain.

The *main contributions* of this work are summarized as follows:

- We conduct an SLR to identify the commonly reported quality attributes and design decisions related to the architectural design of IoT systems supported by blockchain.
- We provide a catalog of relevant architectural tactics that could assist software architects and engineers with different options to design this category of systems and improve their decision-making options. Such catalogs, e.g., References [8, 11] have already shown their value in aiding practicing architects in both design and analysis.
- We identify potential areas for future research that include architectural support for specific quality attributes, empirical research to evaluate the impact of the identified quality attributes, and research effort to explore tradeoffs among the quality attributes and identified tactics.

The remained of this article is organized as follows. Section 2 summarizes previous efforts on architecting IoT systems supported by blockchain. Sections 3 and 4 describe the research method and the analysis of the primary studies respectively. Section 5 presents the architectural tactics for the design of IoT systems supported by blockchain. Section 6 discusses the main findings and potential areas for future research. Section 7 and 8 include the identified threats to the validity of our study and conclude the article respectively.

2 RELATED WORK

In this section, we present relevant studies on the adoption of blockchain in IoT systems and fundamental work on architecting IoT systems supported by blockchain. Our review devotes the most attention to work closely related to the identification of architectural tactics for IoT systems supported by blockchain. There have been several studies that have attempted to analyze blockchain as a potential technology to solve security issues in IoT systems. Unfortunately, these studies assess the integration of blockchain in IoT systems from an application perspective without considering the quality attributes and design decisions that can impact the architectural design of this category of systems. Our work mainly differs from the existing studies on the integration of blockchain in IoT systems as follows: First, we conduct an SLR to investigate the commonly reported quality attributes and design decisions to be considered when architecting IoT systems supported by blockchain. Our findings are drawn from 100 research papers that are selected from a set of 575 relevant publications on IoT and blockchain. Second, we focus particularly on the extraction of architectural tactics for the design of IoT systems supported by blockchain and describe them using the template suggested by Lewis et al. [35]. Third, we identify potential areas for future research that include architectural support for specific quality attributes, empirical research to evaluate the impact of the identified quality attributes, and research effort to explore tradeoffs among the quality attributes and identified tactics.

2.1 Surveys in IoT and Blockchain

Recently, Conoscenti et al. [13] conducted a comprehensive systematic literature review to study the application of blockchain technology and its benefits in terms of decentralization and security. The study describes several use cases where data storage management, trade of goods and data, and identity management have been identified as potential IoT cases to be enhanced with blockchain. Moreover, Christidis et al. [12] emphasized the advantages and disadvantages of adopting blockchain in IoT systems and the use of smart contracts for data sharing and autonomous governance. Yeow et al. [78] critically reviewed the decentralized consensus systems for architecting edge-centric IoT systems by focusing on the data structure, consensus protocols, and transaction models. In addition, Fernández-Caramés et al. [20] presented a review on the impact of blockchain in IoT and the current challenges regarding the design, development, and deployment of IoT systems supported by blockchain. This review also identifies gaps in the literature that can guide researchers and practitioners on the design of future blockchain-based IoT systems. Reyna et al. [52] discussed the benefits and challenges of the integration of blockchain and IoT and the recent platforms and applications for combining these technologies. This survey also presents three architectures for facilitating the communication between IoT devices and blockchain. Moreover, Ali et al. [2] presented a comprehensive survey to investigate the current efforts for the integration of blockchain and IoT and summarize some solutions to enhance data privacy, security, identity management, data management, and monetization in IoT systems. Similarly, Panarello et al. [46] carried out a systematic survey to analyze the current research efforts on the use of blockchain in IoT applications by categorizing the existing literature based on different domains. In addition, the survey describes the challenges and future research directions for realizing the adoption of blockchain in IoT systems. In another work, Ferrag et al. [21] presented a survey on the current effort trends and challenges in the integration of blockchain in IoT systems by providing an overview of the use of blockchain in different IoT domains (i.e., Internet of Vehicles, Internet of Energy, Edge Computing). Hong-Ning et al. [14] conducted a survey in IoT and blockchain with a special focus on the challenges in IoT, an overview of the blockchain technology, and the main opportunities of integrating both technologies. In particular, the authors summarize the main IoT applications supported by blockchain and the key role of the 5G-beyond networks in the convergence of IoT and blockchain. Moreover, Mingli et al. [66] proposed a systematic survey of blockchain and its application in IoT where the fundamental issues and the open challenges on the integration of both technologies are discussed. In particular, the authors analyze the blockchain architecture with a special focus on the adoption of blockchain in other areas (i.e., Artificial Intelligence and Edge Computing). In contrast to the above, our study explicitly defines which are the commonly reported quality attributes in the literature considered in the design of IoT systems supported by blockchain. Sin Kuang et al. [39] presented solutions for the integration of blockchain with IoT. Even though the majority of surveys mainly focus on the advantages of integrating blockchain and IoT in terms of decentralization, security, and data privacy, our findings are drawn from 100 research publications on blockchain and IoT to identify the most commonly reported quality attributes and design decisions that need to be met when integrating these two technologies.

2.2 Fundamental Work on the Integration of Blockchain and IoT

Lee et al. [34] presented a secure and scalable firmware update scheme based on blockchain where IoT devices first need to compute the hash of the downloaded file to check its integrity. To reduce the computational load and data storage requirements in blockchain, the system relies on a P2P network where the firmware updates are spread across multiple nodes to ensure their availability. Moreover, Dorri et al. [17] proposed a lightweight blockchain with three-layer architecture: smart home (centrally managed), overlay network (public blockchain), and cloud to improve the security and privacy of smart homes. The system implements a distributed trust model in the overlay network to reduce the processing overhead and energy requirements of Proof-of-Work (PoW) consensus. An alternative way is relying on edge computing to shift computation and data storage requirements to powerful IoT devices to minimize latency and improves the scalability of the blockchain network as suggested by Stanciu [59]. Similarly, Bahga et al. [5] presented a decentralized and trusted platform called BPIIoT for enabling powerful devices to communicate and managing the manufacturing resources in a P2P network. This system relies on an intermediary component acting as a one-to-one proxy to facilitate the communication between the IoT node and secure communication between them. A different approach is suggested by Shabandri et al. [55] where the system relies on Tangle structure instead of blockchain to improve scalability and reduce latency in transaction confirmation. Only a few attempts have been identified in the literature regarding the integration of blockchain and IoT from the software architecture perspective. Among the existing works, Liao et al. [37] proposed a taxonomy to capture the most significant architectural issues in blockchain-based systems and their impact on the non-functional requirements. Similarly, Liao et al. [38] identified the architectural design issues for architecting IoT systems supported by blockchain that include location of the blockchain nodes, distribution of logic, and data, and integration mechanisms. Based on these design decisions, this study proposes four architectural styles: fully centralized, pseudo-distributed, distributed, and fully distributed. In another work, Reyna et al. [52] emphasized three alternatives for enabling the interaction between IoT devices and blockchain including IoT-IoT, IoT-blockchain, and hybrid approach. Xu et al. [71] described a set of architectural patterns for blockchain-based applications that include *External world patterns, data management patterns, security patterns, and contract structural patterns.* Similarly, Eberhardt et al. [18] proposed five patterns regarding on-chain or off-chain data called *Challenge response pattern, Off-chain signatures pattern, Content-addressable storage pattern, Delegated computation pattern, and Low contract footprint pattern.* In another work, Wessling et al. [61] presented a set of architectural tactics for building blockchain systems by comparing two variants of an Ethereum smart contract implementation.

3 RESEARCH METHODOLOGY

We conduct an SLR to identify the most commonly reported quality attributes and design decisions to consider when architecting IoT systems supported by blockchain. Bass et al. [8] have defined a quality attribute as "a measurable property of a system to evaluate how well they satisfy the business goals." We use this definition to identify the software quality attributes of IoT systems supported by blockchain and reason about their significance and application in this category of systems. We then extracted from the literature a catalog of relevant architectural tactics for IoT systems supported by blockchain and analyzed their role in realizing the identified quality requirements. We followed the SLR guidelines suggested by Kitchenham et al. [33] and Petersen et al. [47], which include (i) research questions, (ii) search strategy, (iii) inclusion and exclusion criteria, (iv) study selection, and (v) data extraction and synthesis procedures.

3.1 Research Questions

This study aims to identify the most commonly reported quality attributes (RQ1) and architectural tactics (RQ2) for realizing the desired quality attributes within the system.

- *RQ1: Which are the most commonly reported quality attributes in the literature for the architectural design of IoT systems supported by blockchain?*
  Aim: Identify the most commonly reported quality attributes in the reviewed literature that need to be addressed when architecting IoT systems supported by blockchain. *Relevance:* By answering this research question, we can help architects and designers with (i) an understanding of the quality attributes to consider when architecting this category of systems and (ii) an overall view of the possible tradeoffs among these quality attributes.
- *RQ2: What are the relevant architectural decisions, strategies, and tactics for achieving the desired quality attributes in IoT systems supported by blockchain?*
  Aim: Investigate the architectural tactics, styles, views, patterns, models, and design decisions for IoT systems supported by blockchain and how they can influence in realising quality requirements of interest. *Relevance:* The result of this research question can help architects and designers with (i) an understanding on the architectural design decisions that are used to achieve particular quality attributes and (ii) control the quality attribute model through architectural decisions to achieve a desired response.

3.2 Search Strategy

We defined our search strategy according to the practices and guidelines for systematic mapping studies suggested by Kitchenham et al. [33] and Petersen et al. [47]. Since our study focuses on the integration of blockchain and IoT, the scope of this review is restricted to IoT systems supported by blockchain.

- *Search string:* We formulated a search string derived from the proposed research questions that includes the following terms and their synonymous: (i) *IoT*, (ii) *blockchain*, and (iii) *software architecture*. We combined these terms and created the following search string, which was checked against a set of known primary studies to evaluate its reliability.

  (internet of things OR internet of thing OR iot) AND (blockchain OR blockchain technology OR block-chain OR BC OR distributed ledger technology OR DLT) AND (Fog computing OR fog OR edge computing OR edge) AND (software architecture OR software design OR software requirements OR architectural tactics OR architectural styles OR patterns OR reference architectures)

- *Search databases:* We carried out our automatic search on electronic databases and indexing libraries (i.e., IEEE Xplore, ACM Digital Library, Scopus, and Web of Science). These databases were selected based on (i) the variety of electronic resources and library catalogs that they provide to support research in software engineering [32, 47] and (ii) their popularity among systematic mapping studies in software engineering [47]. Our work focused on advanced and high-quality journals, conferences proceedings, and scientific workshops while excluding other sources that provide irrelevant information related to the research questions including books, thesis, talks, blogs, and presentations. In addition, our preliminary search was not restricted to the publication date to enable a broad coverage of studies related to the research questions of interest. We used the Publish or Perish [28] tool to retrieve academic results from the selected digital libraries and keep the metadata for further analysis.

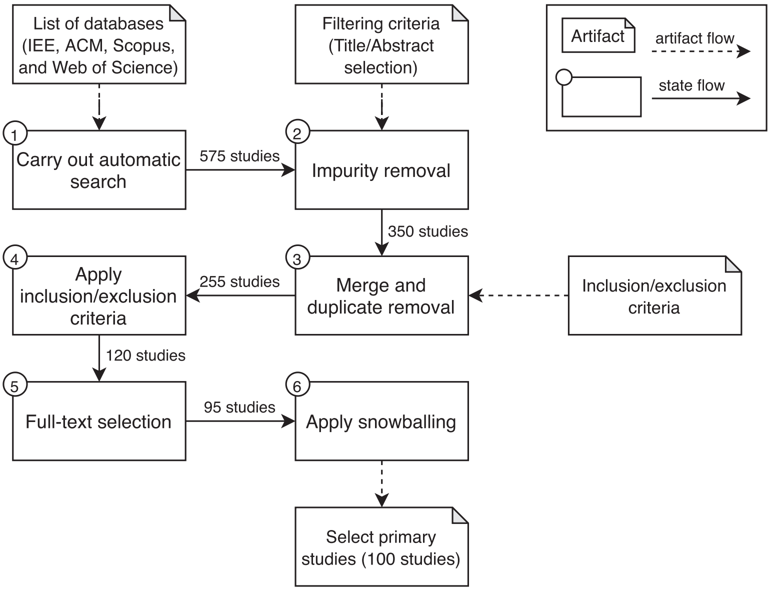
List of databases (IEE, ACM, Scopus, and Web of Science)
Filtering criteria (Title/Abstract selection)
Artifact
artifact flow
state flow
1
Carry out automatic search
575 studies
2
Impurity removal
350 studies
4
Apply inclusion/exclusion criteria
255 studies
3
Merge and duplicate removal
Inclusion/exclusion criteria
120 studies
5
Full-text selection
95 studies
6
Apply snowballing
Select primary studies (100 studies)

Fig. 1. Overview of the selection process.

3.3 Study Selection

We defined a study selection procedure to identify the publications that provide direct evidence about the proposed research questions. This procedure was discussed and revised by the supervisor, team members of the Software Engineering group, and experts in the field. Figure 1 shows the number of studies included and excluded in the selection process.

(1) *Initial search:* We retrieved a total of 575 relevant studies from the four selected databases using the designed search string.
(2) *Impurity removal:* We manually removed studies based on titles and abstracts that were not relevant to the proposed research questions. In particular, we read the abstract to decide whether a study needs to be maintained for the next round of selection. This process is conducted by the first author and the second author, resulting in 350 of 575 initial studies.
(3) *Merging and duplication removal:* The selected studies were analyzed by the first author and the second author to remove duplicates, since some Scopus publications are also available in IEEE Explorer and ACM. As result, we created a single dataset of 255 studies to be used in the next round of our study.
(4) *Selection criteria:* Table 1 describes the inclusion and exclusion criteria that were applied by the first author and the second author to all selected studies and resulted in a total of 120 studies.

    Even though the exclusion criteria E2 in Table 1 removes the secondary studies, we still considered them to (i) identify their contribution to the study, (ii) define as many studies related to the research questions, and (iii) investigate their relevance of our study to the current research on IoT and blockchain.
(5) *Full-text selection:* We performed a full-text reading of the selected studies to ensure their alignment with the research questions, which reduced the number of candidate studies to 95. All the selected publications were inspected by the first author and the second author

Table 1. Inclusion and Exclusion Criteria

#	Inclusion criteria
I1	Discuss the quality attributes requirements that are considered when integrating blockchain and IoT.
I2	Provide software architecture solutions including styles, tactics, patterns, views, or reference models for the design of IoT systems supported by blockchain.
I3	Support evaluation of the architectural strategies, methods, or techniques for the integration of blockchain and IoT (e.g., case scenarios, prototype solutions, simulations, etc.).
I4	Subject to peer-review.
I5	Written in English.
#	**Exclusion criteria**
E1	Propose the integration of blockchain and IoT, but do not present an architecture.
E2	Include blockchain and IoT as secondary studies (e.g., systematic literature review, surveys, etc.).
E3	Present as tutorial papers and editorials that were not in the form of a published paper that does not provide direct evidence of the integration of blockchain and IoT.
E4	Full papers that document the approach and provide potential evaluations.

Table 2. Data Extraction Form

Data item	Value	RQ
Study ID	Number	
Study title	Name of the study	
Author name	Author(s) in the study	
Publication year	Number	
Publication type	Conference, journal, and workshop	
Quality attributes	The quality attributes identified in the studies.	RQ1
Architectural design	Design decisions towards the integration of blockchain and IoT.	RQ2

to eliminate bias in our study, we shared the final set of studies with software engineering researchers for evaluation.

(6) *Snowballing:* We complemented our full-text reading with recursive backward and forward snowballing activities described by Wohlin [62] to complement the automatic search. In the backward snowballing, we focused on the references of the primary studies while in the forward snowballing, we used Google Scholar to obtain new publication results. As a result, a total of 5 studies were added to the final set that were assessed using the inclusion and exclusion criteria.

3.4 Data Extraction

We designed a framework to rigorously extract all the information needed from the primary studies for addressing the proposed research questions. Table 2 describes the data extraction fields with their corresponding values and related research questions. Since the first author performed the data extraction procedure, the second author selected a random sample of the primary studies to cross-check the results with those of the first author to reduce the threads to the reliability (see Section 7).

3.5 Categorization of the Architectural Decisions

An architectural decision should be accompanied by the rationale for the decision, couched in terms of how this decision helps to achieve one or more desired quality attributes, along with any drawbacks or tradeoffs [8]. Table 3 summarizes the main blockchain-related design decisions identified in the primary studies and their impact on the desired quality attributes [73].

4 ANALYSIS OF THE PRIMARY STUDIES

In this section, we report the analysis results of the primary studies to identify the quality attributes and design decisions to be considered when architecting IoT systems supported by blockchain.

4.1 Quality Attributes for Architecting IoT Systems Supported by Blockchain

Table 4 presents the commonly reported quality attributes to consider when architecting IoT systems supported by blockchain (for detailed explanations, please see Section 3) and some examples. It is worth noting that not all the studies have explicitly mentioned what quality attributes they address to realise the system functionality. We identify them by looking at the primary studies in detail and relate their solution to the users and system requirements. In addition, some studies focus on achieving more than one quality attribute. For instance, [73] considers performance and security as the the most important quality attributes for this category of systems while [16] highlights security as a critical quality requirement. In the reviewed literature, we identify security, scalability, and performance as the commonly reported quality attributes with a total of 55, 23, and 18 studies respectively (see Table 4). In addition to these quality attributes,there are other quality requirements that appear in a few studies such as interoperability, mobility, adaptability, and efficiency, which could also have a significant impact on the architectural design of IoT systems supported by blockchain. For each quality attribute, we provide a brief explanation and motivation for their importance in IoT systems supported by blockchain as follows:

4.1.1 Security. According to Barbacci et al. [7], security mainly comprises three concerns: confidentiality, integrity, and availability. Confidentiality refers to protect data from unauthorized disclosure while integrity prevents unauthorized data modification. Similarly, availability ensures data access to authorized users. Thus, an ideal IoT system supported by blockchain must implement access control permissions via smart contracts to restrict access to only authorized participants (confidentiality) and keep critical data and hashes of raw data in the blockchain to ensure its immutability and integrity (integrity). Moreover, this category of systems must replicate sensor data across the P2P network to ensure its availability to authorized participants (availability) [42, 52].

4.1.2 Scalability (Concerning Blockchain Size and Transaction Throughput). An optimal IoT system supported by blockchain must ideally achieve a low transaction throughput with the increase in the number of miners and validator nodes in the blockchain network. However, the rise in the number of blockchain nodes could increase the number of transactions, thus increasing the size of the blockchain. With the increasing size of the blockchain, the storage requirements also increase. It could put more limitations on the integration of resource-constrained IoT devices to act as miner nodes in the blockchain network. In addition, the increased size of the blockchain could result in longer synchronization for new devices or users who want to join the blockchain network [42, 53].

4.1.3 Performance (Concerning Latency in Transaction Confirmation). An ideal IoT system supported by blockchain must achieve low latency in transaction confirmation to ensure instant consensus agreement, which is a fundamental requirement in the majority of real-time IoT systems such as smart vehicles, smart grids, and intelligent transportation systems. A possible way to

Table 3. Design Decisions for Blockchain-based Systems

Design decision	**Quality attribute and tradeoffs**	**Impact**
Data storage and computation: What data and computation should be placed on-chain and off-chain?	*On-chain:* Enhances security of IoT data, but it is computational expensive and energy hungry.	Limit the amount of data that can be stored on-chain.
	Off-chain: Improves scalability and availability of the blockchain, but it represents high maintenance cost and requires additional trust.	Interaction issues between on-chain and off-chain storage.
Blockchain scope: What type of blockchain should be used?	*Public:* Ensures data transparency and auditability, but potentially poor performance (i.e., high transaction confirmation cost and limited block size).	Privacy and confidentiality concerns, since data are available to all blockchain nodes.
	Private: Improves performance of the blockchain network, but it offers little support for data auditability and transparency.	Centralization issues, since data are managed by a single entity.
	Consortium: Managed by multiple organizations and ensures a better performance, scalability, and security.	Has the same advantages of a private blockchain but operates under leadership of a group.
Consensus protocols: Which consensus protocol should be selected?	*PoW:* Computationally expensive and time-consuming.	Requires powerful hardware for mining transactions.
	Proof-of-Stake (PoS): Improves performance and requires less computation and energy power, but the extensive control and authority over technical and economic aspects by participants could lead to a monopoly problem.	Centralisation of voting power results in control of the blockchain network.
	Practical Byzantine Fault Tolerance (PBFT): Enhances security and performance, but it impacts scalability.	Single-point-of-failure due to the size of the blockchain network.
	Proof-of-Authority (PoA): Improves the security, because an authority is assigned a fixed time slot within, which it can generate blocks.	Assume trusted authorities.
Blockchain data structure: Which type of data structure should be configured?	*Single chain:* Easy chain management and permission control, but it makes complex data management.	With the increasing number of transactions from IoT devices, a single blockchain might become overloaded and make difficult data retrieval.
	Multiple chains: Easy data management, but it makes harder chain management and permission control.	Allows recording IoT data in different blockchains for easy data storage and retrieval.
Blockchain deployment: Where the blockchain should be deployed?	*IoT:* Improves scalability of the blockchain, but it leads to performance issues.	Enable IoT devices to work as nodes of the blockchain network.
	Fog: Improves scalability and performance of the blockchain network, but leads to management issues.	Ensures decentralization in the end-to-end system.
	Cloud: Ensures decentralization and improves security in the cloud, but leads to high latency and bandwidth consumption.	Enable large amount of computing resources.

Table 4. Quality Attributes

Quality attribute	% of studies	Representative examples
Security	55%	[PS1, PS3, PS5, PS6, PS7, PS10, PS11, PS12, PS15, PS18, PS19, PS21, PS23, PS24, PS25, PS27, PS28, PS29, PS30, PS32, PS33, PS34, PS35, PS36, PS39, PS40, PS41, PS45, PS47, PS48, PS49, PS51, PS52, PS55, PS57, PS58, PS62, PS65, PS66, PS69, PS70 ,PS72, PS75, PS77, PS78, PS80, PS81, PS85, PS86, PS88, PS90, PS91, PS93, PS95, PS99]
Scalability	23%	[PS5, PS7, PS11, PS12, PS15, PS19, PS26, PS27, PS29, PS30, PS32, PS34, PS36, PS38, PS40, PS42, PS43, PS48, PS49, PS52, PS60, PS61, PS6]
Performance	18%	[PS2, PS23, PS25, PS37, PS38, PS41, PS50, PS56, PS57, PS63, PS67, PS68, PS76, PS79, PS82, PS86, PS92, PS93]

minimize transaction confirmation time while achieving the same level of security, consists of reducing the block generation time, but it could require to wait for more confirmations due to the less difficulty in mining a block. The latency could also be reduced by increasing the block size. For instance, in Bitcoin blockchain, the block size can be increased from 1 to 2 MB to improve the throughput in the network, but it will lead to longer blocks that could be difficult to be propagated in the blockchain network. Moreover, the increased block size will result in a continuous increase of blockchain size, which results in more full nodes with high storage capacity to store a copy of the complete blockchain [42].

4.1.4 Interoperability. An ideal IoT systems supported by blockchain must ensure data exchanges between different blockchain implementations and the integration of heterogeneous devices as blockchain nodes [3, 52]. Specifically, multiple blockchains can be used to enable separation of concerns among different type of transactions and business goals, but their interaction needs to be guaranteed to meet the requirements of IoT systems [73]. In addition, IoT devices working as full or lightweight blockchain nodes should be able to communicate and share information with nodes in another chain [57].

4.1.5 Efficiency. An optimal IoT system supported by blockchain must ensure a cost-effective and efficient utilization of hardware and power resources in IoT devices and blockchain nodes [20, 53]. On one hand, the reduction of redundant data movements from IoT devices to the cloud could minimize latency and energy consumption in the system [52]. On the other hand, the selection of resource intensive consensus protocols like PoW could impose new challenges in the adoption of blockchain in IoT systems due to the constraint resources in the majority of IoT devices. Thus, a lightweight consensus protocol and an alternative verification mechanism could be required to have a small footprint and low energy costs [17].

4.1.6 Adaptability. An IoT system supported by blockchain must adapt IoT networks and rules in smart contracts based on the user's and system requirements. Specifically, adaptability in IoT refers to dynamic traffic in IoT networks and heterogeneous features in IoT devices (i.e, different software and hardware resources) that allow them to join and leave the network [50]. It makes easier for attackers to compromise IoT devices with fake ids and manipulate IoT networks in the presence of such networks. Thus, IoT networks need to continuously adapt to changes in traffic load and uncertainties in the environmental conditions. For the blockchain, adaptability means changes in the business logic (i.e., rules and agreements) on-chain stored in the smart contracts

Table 5. Distribution of Computation and Storage

Design decision	Option	Representative example
On-chain	Transactions and smart contracts	[PS1, PS5, PS7, PS12, PS13, PS16, PS22, PS25, PS26, PS27, PS29, PS30, PS31, PS32, PS34, PS35, PS37, PS38, PS39, PS40, PS42, PS45, PS47, PS49, PS51, PS52, PS53, PS55, PS56, PS60, PS62, PS68, PS71, PS72, PS74, PS75, PS77, PS78, PS79, PS81, PS82, PS88, PS91, PS93, PS95, PS97, PS98]
On-chain/Off-chain	Transactions and smart contracts/Cloud, local database, and P2P storage	[PS18, PS70, PS43, PS44, PS57, PS66, PS69, PS83, PS87]

based on the environmental context [40]. However, if the blockchain is mainly as a secure storage, then the adaptability in smart contracts do not need to be ensured.

4.1.7 Mobility. An ideal IoT system supported by blockchain must be able to handle the mobile aspect of the majority of sensors and IoT devices that change their locations based on the hardware resources and system requirements. Similarly, mobility in blockchain means having intermediate energy distributors, analytic or storage to reduce computation and storage loads in blockchain nodes and improve energy efficiency in this category of systems [42].

4.2 Architectural Decisions to Consider in IoT Systems Supported by Blockchain

The commonly reported architectural decisions for designing IoT systems supported by blockchain as defined earlier in Section 3 are summarized as follows:

4.2.1 Distribution of Computation and Storage. One of the major design decisions when architecting IoT systems supported by blockchain is what data and computation should be kept on-chain or maintained off-chain [37, 40]. On the one hand, this decision should consider the limited computation (transaction throughput) and data storage space (block size) in public blockchains [73]. For instance, Bitcoin has a block size of 1 MB and can only handle seven transactions per second (tps) on average, while VISA can perform 60,000 tps on average. In addition, the use of blockchain storage has a high cost compared to the use of conventional storage systems (i.e., local database, cloud or P2P storage) [29]. On the other hand, the use of blockchain storage should consider that data are replicated across the blockchain nodes.

Many studies in the reviewed literature store hashes of raw data in blockchain or record IoT data in smart contracts while other studies use an off-chain storage to record raw data generated by IoT devices. Table 5 shows the distribution of IoT data on-chain and off-chain along with some representative studies under each category.

On-chain. A common practice for data management in blockchain-based systems is to store small critical data, hashes of the raw data, and metadata in the blockchain [40]. These data can be packed into (i) a transaction or (ii) a smart contract.

- **Recording data as transaction:** Due to the limited data storage space in blockchain, a small amount of data can be stored on-chain or as part of a transaction [40, 73]. These systems include Blockchain for data sharing [PS9], Blockchain Transportation [PS1], IoT updates [PS7], Optimized blockchain [PS2], and MeDShare [PS4]. MediChainTM [PS50] is a special case, because it stores metadata and hashes of the raw data on-chain to ensure its integrity and immutability. However, many of these systems do not explicitly mention what data are stored on-chain, which imposes new challenges in the development of future IoT systems supported by blockchain.

- **Recording IoT data via smart contracts:** A smart contract is a general program that can codify the states of physical assets or data exchanges among IoT devices [37]. However, the storage of large amount of logic or data in the blockchain could lead to high transaction throughput, since the majority of blockchain nodes need to reach consensus to validate transactions. The following are examples of studies using on-chain data storage through smart contracts, i.e., Blockchain Transportation [PS1], MIoT [PS39], Auth IoT [PS61], and MediChain [PS50]. However, the use of smart contracts in blockchains has a deployment and execution cost that need to be considered when designing and architecting IoT systems supported by blockchain.

Off-chain. The raw data, source of data requests, smart contract addresses, and code of smart contracts are kept in an off-chain storage (i.e., local database, cloud, or P2P storage) due to the limited data storage space in public blockchains [56]. These storage solutions have their own advantages and disadvantages in terms of transparency, cost of storage, and centralization. The following are a set of studies that rely on cloud platforms as off-chain storage, i.e., Optimized blockchain [PS2], Blockchain for data sharing [PS9], Vegvisir [PS11], and IoT data assurance [PS18]. MediChain [PS50] is a special case, because it encrypts sensitive data (i.e., diagnostic images, lab test results, prescript, treatment plans) before storing it in a remote resource (i.e., enterprise cloud or data center), which are located in a multi-hop proximity of the IoT devices. Similarly, MeDShare [PS4], Forensic SDN [PS36], and Hybrid-IoT [PS43] use a local database located in single-hop proximity of IoT devices as off-chain storage. The last set of systems use a P2P storage to record IoT data and include Blockchain auditable storage [PS8], IoT protection-blockchain [PS14], and Emergency SH [PS33].

4.2.2 Blockchain Configuration. This comprises a set of design decisions (i.e., type of blockchain, consensus protocol, and data structure) to consider when deploying blockchain-based systems. Table 6 summarizes the results of the blockchain configuration design decisions with some representative examples under each category in detail.

Type of blockchain. (referring to the use of a public or private blockchain [70]). In a *public blockchain*, anyone can join the network and perform transactions, which could enhance transparency, but it could lead to user anonymity and data privacy issues. Moreover, public blockchains have low transaction throughput by design because of the delay in final transaction confirmation, especially in PoW-based blockchains [42]. The majority of systems in the studies use Ethereum platform to facilitate the deployment of blockchain-based IoT systems. These systems include Blockchain auditable storage [PS8], Hybrid BC-IoT [PS12], Privacy SH [PS22], Integrity CPS [PS24]. However, a *private blockchain* could be managed and hosted by a single organization that defines who can join the blockchain network, thus limiting the number of miner nodes. In addition, private blockchains restrict user with access to only the transactions corresponding to them, which enable competing organizations to keep the privacy and confidentiality of their transactions, as in the case of Hyperledger [23]. The next set of systems uses a private blockchain (i.e., Hyperledger Fabric and Multichain) to support different IoT use cases as follows: Blockchain for Edge [PS6], IoT updates [PS7], Blockchain for data sharing [PS9], and Blockchain as a Service for IoT [PS17]. All these systems assume that a private blockchain is required to securely record IoT transactions and guarantee their privacy. Similarly, a *consortium blockchain* is a hybrid blockchain with public and private blockchain features that is maintained by a group of organizations. Each organization keeps a mining node in the blockchain network and validates a block when the

Table 6. Design Decisions for Blockchain-based Systems

	Type of	Consensus	
Data structure	**blockchain**	**Protocol**	**Representative example**
Blockchain	Public	PoW	**Ethereum (40):** [PS5, PS12, PS14, PS16, PS19, PS20, PS22, PS23, PS24, PS26, PS27, PS30, PS33, PS39, PS40, PS43, PS49, PS52, PS53, PS57, PS61, PS64, PS65, PS66, PS68, PS70, PS71, PS73, PS75, PS79, PS82, PS83, PS85, PS87, PS88, PS90, PS91, PS94, PS95, PS99]
			Bitcoin (9): [PS2, PS8, PS23, PS35, PS36, PS74, PS80, PS81, PS92]
		PoS:	**Monax (3):** [PS13, PS40, PS49]
	Private	Byzantine Fault Tolerance (BFT)	**Hyperledger Fabric (15):** [PS6, PS9, PS25, PS28, PS34, PS38, PS44, PS45, PS46, PS50, PS51, PS58, PS93, PS97, PS100]
		Round Robin	**Multichain (8):** ¨[PS7, PS17, PS37, PS62, PS76, PS77, PS78, PS96]
		Proprietary protocol	**Proof-of- Service (1):** [PS5]
			Proof-of-Inclusion (1): [PS27]
			Proof-of-Authority (1): [PS32]
DAG	N/A	IoTA	[PS11, PS28, PS48, PS60]

majority of nodes agrees on the transaction. Even though the mining nodes can read all the transactions in the blockchain network, this access can be restricted to specific nodes, which could result in a possibility of tampering due to the increased centralization [83].

Data structure. (referring to the representation of the transactions in the distributed ledger). The data structure consists of a chain of blocks connected to each other where transactions are stored in a chronological order [2]. In addition, the transactions are replicated over all blockchain nodes and bundle into a block for validation, which impacts scalability and performance of the blockchain network [53]. To improve the scalability of blockchain, Direct Acyclic Graph (DAG) has been proposed in the literature as another type of distributed ledger composed by a network of nodes connected to each other that confirm transactions [55]. Even though the majority of studies use blockchain as data structure, e.g., IoT authentication [PS13], Opportunistic IoT [PS49], Privacy SH [PS22], Integrity CPS [PS24], a few studies use DAG, i.e., Vegvisir [PS11], P2P data monetization [PS28], DAG [PS48], and Privacy distributed [PS60].

Consensus protocol. It is the procedure used by all blockchain nodes to reach a common agreement on the state of the distributed ledger. The most used consensus protocols are PoW, Proof-of-Stake (PoS), Practical Byzantine Fault Tolerance (PBFT), and Proof-of-Authority (PoA). The selection of a consensus protocol could have a high impact on the security and scalability of IoT systems supported by blockchain.

4.2.3 Deployment of Blockchain. The location of the blockchain nodes is a fundamental design decision to consider when deploying blockchain-based systems, since it has an impact on some quality attributes (i.e., performance, and scalability) [37, 73]. The most common practice for integrating blockchain and IoT is leveraging blockchain on (i) edge computing, (ii) cloud provided by a third party, or (iii) local network [38, 70]. The majority of the primary studies deploy blockchain in a local network to minimize latency and guarantee privacy of IoT transactions. The next set of studies implement a blockchain in the cloud infrastructure, which results in high latency and bandwidth consumption. Other set of studies use edge platform to deploy a blockchain network where edge nodes can operate as miners and full nodes. However, the computation and data storage space in edge nodes is still limited and could become a bottleneck in the network as the amount of IoT data increases over time.

5 ARCHITECTURAL TACTICS FOR IOT SYSTEMS SUPPORTED BY BLOCKCHAIN

This section presents the extracted architectural tactics for the design of IoT systems supported by blockchain. First, we highlight the difference between architectural tactics and patterns to provide software architects with an in-depth understanding of their impact on the architecture design of a system. According to Harrison and Avgeriou [27], *architectural tactics* are "design decisions that influence the control of individual quality attribute requirements", while *patterns* "describe the high-level structure and behavior of a software system as the solution to recurring problems." For instance, a design decision concerning security could be how to prevent attacks on the system?. A possible tactic to improve security could be authentication of users [27]. It is worth noting this work focuses on the extraction of architectural tactics from the reviewed literature to satisfy particular quality attributes of IoT systems supported by blockchain and to provide different options for the architectural design of this category of systems. In this context, Bass et al. [8] present a list of architectural tactics to meet the following quality attributes: availability, interoperability, modifiability, performance, security, testability, and usability. We examine whether these tactics have been applied or adjusted in the context of blockchain and IoT systems to deliver a catalog of relevant architectural tactics for IoT systems supported by blockchain. Our work elicits the tactics from the primary studies based on (i) the explicitly stated quality attributes, (ii) inferred quality attributes from the primary studies, (iii) the commonly reported blockchain-based design decisions, and (iv) common components and their relations across the selected studies. Specifically, we couple a quality attribute to relevant design decisions and translate them into architectural tactics. In addition, we rely on a surrogate component widely used in cyber-foraging systems to offload computation or data to more powerful devices [35]. In most proposed tactics, the surrogate acts as an intermediate between IoT devices and the blockchain and is used to collect sensor readings from resource-constrain devices, which cannot directly to a blockchain network and perform mining tasks. In other cases, IoT networks comprise powerful devices that can connect directly to the blockchain without the need for a surrogate component. Table 7 shows a catalog of architectural tactics for security, scalability, performance, and interoperability. We report each tactic using the template described by Lewis and Lago [35] as follows:

- *Summary:* Brief introduction of the tactic.
- *Motivation:* Rationale behind the implementation of the architectural tactic.
- *Description:* Detailed explanation of the components in a tactic and their interaction to achieve a particular quality attribute.
- *Constraints:* Benefits and drawbacks of applying the tactic.
- *Example:* Application of the tactic on the existing literature.
- *Related tactic:* Relation with other tactics to achieve its potential.
- *Variations (optional):* Slight modification of the tactic from its original form to optimize it.

Even though the same diagram style was used to describe most of the architectural tactics, slight modifications on the diagram were required to understand some of them.

5.1 Encryption of On-chain Data

Summary: Encrypt IoT data before sending it as transactions to a blockchain to ensure their integrity and immutability.

Table 7. Architectural Tactics for IoT Systems Supported by Blockchain

Quality attribute	Tactic name	Description
Security	(1) Encryption of on-chain data	Encrypt IoT data before sending transactions to the blockchain to ensure its confidentiality and privacy.
	(2) Access permission via smart contracts	Enable access control to IoT data through smart contracts.
	(3) Two-authentication factor for IoT devices	Enable an additional layer of security to authenticate IoT devices.
	(4) Trusted blockchain nodes	Ensure integrity of data and IoT devices by identifying and authenticating them in the blockchain network.
Scalability	(5) Off-chain data storage	Use a third-party offline data storage for IoT raw data while keeping a digital hash of critical data on-chain for verification.
	(6) Sidechain	Improve scalability of the blockchain by relying on child chains connected to a parent chain.
	(7) IoT devices as lite blockchain nodes	Connect resource-constraint IoT devices to the blockchain network through powerful IoT devices.
	(8) IoT devices as full blockchain nodes	Use powerful IoT devices as full blockchain nodes.
Performance	(9) Caching offload	Use a cache system to offload a subset of data and make further data request faster.
	(10) Surrogate computation	Delegate computation-intensive tasks to edge servers to reduce computation and data storage load in blockchain nodes.
	(11) Sharding	Increase transaction throughput confirmation in blockchain networks.
Interoperability	(12) Two-layer blockchain	Enhance interoperability of public and private blockchains by introducing a two-layer blockchain architecture.

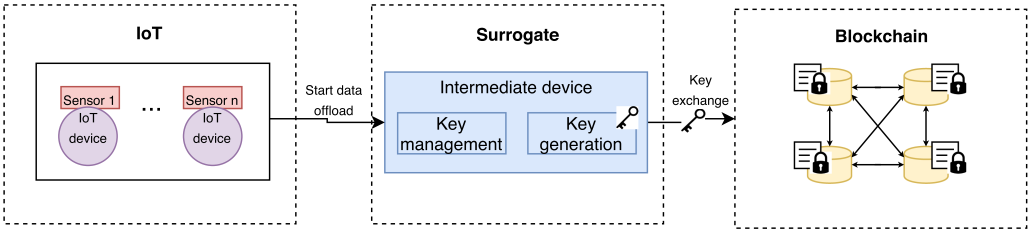

Fig. 2. Encryption of on-chain data where a surrogate device handles the encryption key.

Motivation: One of the main issues in public blockchains is the lack of privacy, since anyone on the Internet can join the network without permission [70]. As result, all the transactions on blockchain are available to everyone in the network and almost every participant has a copy of the entire chain [73]. Therefore, IoT data cannot be deleted or altered in the blockchain network, which leads to better transparency and auditability but impacts privacy and confidentiality in IoT systems.

Description: Figure 2 shows the main components of the encryption of on-chain data tactic. This tactic requires encrypting IoT data to enhance its security before replicating it across the blockchain nodes. A possible way to encrypt and decrypt data using asymmetric cryptography is described as follows [42]. First, one of the nodes in the blockchain creates a public key and shares it during an initial key exchange. Next, if a user wants to send data to a blockchain, then he or she encrypts such data with the public key of the participant who is allowed to view the data. The participant in possession of the corresponding private key can then decrypt the data.

Consequences:

Benefits:

- *Confidentiality:* IoT data in a public blockchain is not in plain text instead it is encrypted with the public key of the authorized participant in a blockchain network and accessible only using the corresponding decryption key.

Drawbacks:

- *Key management and sharing:* The encryption and decryption keys need to be securely shared off-chain and distributed among authorized nodes before submitting any IoT data to the blockchain. If the key management is not handled carefully or shared in a public blockchain, then the encryption keys could be compromised and disclosed. This results in lack of confidentiality and integrity of IoT data stored in a blockchain.
- *Access permission:* Once IoT data has been stored in a blockchain, it is challenging to revoke read access, since blockchain ensures immutability by design. Thus, a participant in a blockchain network can access to encrypted data as long as he or she is in possession of the corresponding decryption key.
- *Data immutability:* Even when IoT data recorded in a blockchain remains encrypted, it could be subject to brute force decryption attacks [71]. With the advancements in the quantum technology, current encryption algorithms could become ineffective in the future [31].

Related tactic: Off-chain data storage tactic (Section 5.5).

Example:

- *Optimized blockchain* [PS2]. All transactions performed in the IoT network are signed and encrypted before sending them to a blockchain and becoming available to all blockchain nodes.
- *IoT updates* [PS7]. The system relies on asymmetric encryption using RSA keys for updating signing and encryption to guarantee data confidentiality and integrity of IoT transactions.
- *Blockchain auditable storage* [PS8]. The transactions consist of the ownership of data streams and corresponding access permissions and are encrypted using asymmetric cryptography to guarantee data confidentiality and integrity.
- *BLE-IoT* [PS71]. The gateway encrypts user preference for IoT devices and stores it in the blockchain to ensure its privacy and confidentiality.
- *IoST* [PS77]. The data requests sent to rule-based expert system are encrypted using synchronous AES encryption method before pushing them to a blockchain for immutable storage.
- *IoT privacy* [PS80]. IoT devices manage a public and private key to send encrypted sensor readings to a validator node that logs the received data as data creation events before adding them as encrypted transactions to the sidechain.
- *IoT data assurance* [PS18]. The data collected by drones are encrypted and signed using a public and private key pair to protect its integrity before making it available in the blockchain network. It is accessible only to whom owns the corresponding decryption key.
- *P2P data monetization* [PS28]. The system uses credentials (i.e., certificate and keys) to protect all the messages in the IoT network before recording them in the blockchain.

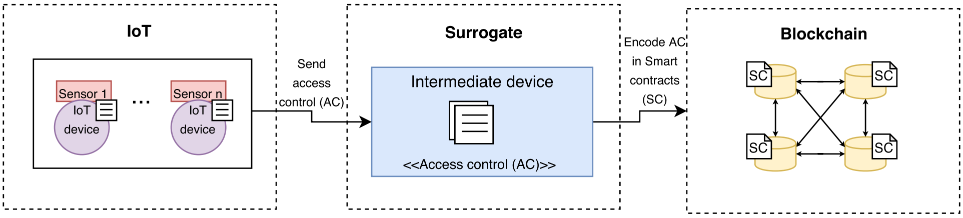

Fig. 3. Access control via smart contract where the surrogate handles the IoT permissions.

- *Blockchain Lightweight IoT Clients* [PS30]. The transactions consist of modification to the account states and are signed using asymmetric cryptography and identified by their hash value, as described in Bitcoin specification.
- *Emergency SH* [PS33]. The asymmetric encryption is used to protect user data and sensitive information from malicious users in the network during data in-transit or in-rest. RSA asymmetric encryption of key length 1024 bits is used to sign the data before pushing it to the blockchain.

5.2 Access Permission Via Smart Contracts

Summary: Enable access control rights to IoT transactions and execute automatic tasks based on pre-defined conditions using smart contracts.

Motivation: Due to the limited computation, storage, and power in sensors and embedded devices, IoT systems rely on cloud services for data processing and analysis. However, data in the cloud can be manipulated and altered by cloud providers [52]. Therefore, blockchain with its smart contracts empowers users with control over their data by restricting access only to authorized blockchain nodes without relying on a cloud service providers [30].

Description: Figure 3 shows the main components of this tactic. The access permission via smart contract tactic requires the deployment of smart contracts on the blockchain network to grant access to IoT data or perform arbitrary computation. The smart contracts can encode fine-grained permissions or contextual policies for sharing services and resources and run as part of the transactions in an autonomous manner [2]. When a user wants to access to a protected resource, he or she has to encrypt the transaction with his public-private key pair and send it to the address of the smart contract in the blockchain. Next, the execution of certain operations in a transaction can be restricted to certain authorized blockchain nodes to enhance security of IoT systems.

Consequences:

Benefits:

- *Security:* Only the blockchain nodes authorized by the smart contracts are able to access to users records without the need of a trusted third party or a cloud service provider for validation or authorization.

Drawbacks:

- *Cost:* If a public blockchain is used to store a smart contract, then the implementation of the access control permissions has an extra cost. This cost includes the deployment and execution of the smart contract in the blockchain network, since each blockchain node must validate it before its approval [42].
- *Flexibility issues:* If no access control is considered initially, then it could be hard to introduce it afterwards due to the structural immutability of smart contracts. Thus, the

implementation of access control via smart contracts can help to deal with changing requirements in the system as long as it is acceptable to have those changes documented as immutable transactions.
- *Codification issues:* The smart contracts should be well written, since once deployed, data stored on them cannot be modified, which can lead to loss of money, wrong decisions, and catastrophic consequences in IoT systems [38].
- *Deployment issues:* Ethereum and Hyperledger Fabric are the most popular blockchain platforms that support smart contracts implementation [53]. However, there are other blockchain platforms that also facilitate the implementation of smart contracts such as EOS, Cardano, Stellar, NEO [42, 60].

Related tactic: N/A
Example:

- *Blockchain manufacturing* [PS76]. The system uses smart contracts to create agreements between users and services providers. These rules are encrypted using symmetric encryption and ensure that only authorized users are able to use services in the network.
- *Bubble of trust* [PS79]. The system uses a smart contract to create agreements between users and services providers, which are encoded as encrypted rules. These rules could be used to ensure that only authorized users are able to use services in the network.
- *Blockchain as a service for IoT* [PS17]. The system implements smart contracts to grant access only to authorized participants in the blockchain network that own the required key. They are able to download and decrypt the protected resources from the blockchain network.
- *MeDShare* [PS4]. The smart contracts are deployed to enable access control policies, data sharing, and revoke access to health data to enhance its security and privacy. In addition, health data provenance and auditing is ensured, since cloud service providers maintain a blockchain network to ensure immutability and data integrity.
- *Privacy SH* [PS22]. A decentralized and scalable access management mechanism is implemented for IoT systems using blockchain. Due to the limited capabilities of the majority of IoT devices, they are connected to a manager node, acting as a lightweight node in the blockchain network. This node defines access control permissions as transactions that are encoded in a single smart contract and executed by the P2P network to make them accessible to all blockchain nodes.
- *Blockchain meets IoT* [PS75]. The system deploys three smart contracts (i.e., access control contract, judge contract, and register contract) to manage access control to the IoT records stored in the blockchain network. The smart contracts enable registering, deleting, and updating the misbehavior-judging methods for managing access control policies. They empower users with control over their own data and facilitates data sharing among trusted participants in the blockchain network.

5.3 Two-authentication Factor

Summary: Enable an additional security layer to the authentication process of IoT devices to ensure integrity and confidentiality of sensor data.

Motivation: With the growing number of IoT devices and the large amount of sensitive and critical data collected by them, security and data privacy become key concerns in IoT systems [4]. However, it is not possible to implement complex security protocols (i.e., encryption and authentication) in IoT devices due to their limited computation, storage, memory, and power lifetime [44]. Although blockchain is envisaged to solve security issues in IoT systems, there is still a need

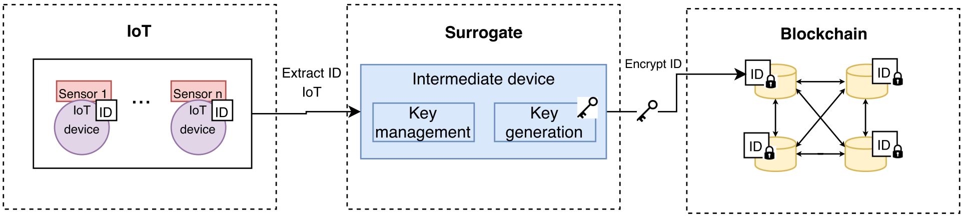

Fig. 4. Two authentication factor where the surrogate records proximity and IoT message exchange.

for developing authentication schemes to protect IoT data in transit where an adversary can take advantage of IoT devices and launch physical or side-channel attacks [6].

Solution: Figure 4 shows the main component of this tactic. The implementation of a two-factor authentication mechanism in resource constrained IoT device requires using an out-of-band channel instead of passwords and shared secret keys [63]. The out-band-channel is provided by manufacturers as first authentication method. This tactic operates as follows. The relationship in terms of proximity and device message exchanges between IoT devices and their verifiers are stored in the blockchain network. If a device is moved beyond its designated verified location, then it is automatically detected and treated as a malicious outsider as the distance relationship is already stored as an immutable transaction in the blockchain. It ensures that only authorized IoT devices can be connected to verifiers and only their transactions can be recorded in the blockchain network.

Consequences:

Benefits:

- *Integrity:* The implementation of a two-factor authentication mechanism enhances the security of IoT devices and protects user's sensitive and critical data from malicious actors. This also ensures that only authorized devices can send transactions to the blockchain, which guarantees the integrity of IoT devices and on-chain data and prevents blockchain nodes from overload.

Drawbacks:

- *Communication complexity:* The use of low-power wireless communication and heterogeneous hardware of IoT devices makes it difficult to create the relationship between an IoT device and its verifier and ensure the integrity of the proximity information between them.

Related tactic: N/A

Example: An example of the application of the two-authentication factor has been identified in IoT authentication [PS13]. This system uses wireless channel characteristics to distinguish between IoT devices at home from outsider devices. Each device needs to authenticate against the verifier device to access services and data in the house. The relationship between the device and its verifier is recorded in the blockchain, which makes it easier to detect when an adversary IoT device wants to gain access to the house.

5.4 Trusted Blockchain Nodes

Summary: Ensure the integrity of sensor data and IoT devices by identifying and authenticating them in the blockchain network.

Motivation: The heterogeneity and dynamic connection of IoT devices (e.g., devices can join and leave the network) make it difficult to assign an ID to identify devices in the IoT network [42]. Before IoT data are sent to the blockchain, its integrity mainly depends on the security of IoT

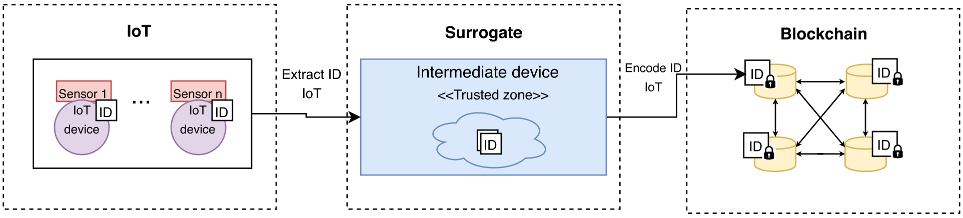

Fig. 5. Trusted blockchain where the surrogate supports the trusted IoT zones.

devices. However, due to their limited computation, storage, and connection lifetime, the majority of IoT devices are vulnerable to attacks that can compromise the devices with fake identifiers to join the network [50].

Description: Figure 5 shows the main components of this tactic. The trusted blockchain nodes tactic relies on the creation of zones where devices can trust and authenticate each other and ensure secure communications and data exchanges [25]. A zone consists of a group of IoT devices managed by a master entity where each device is able to communicate with other devices within its own zone. Any device outside the zone is considered malicious, thus they cannot communicate with other devices and send transactions to the blockchain network. As a result, only authorized IoT devices within a zone can communicate in a secure and transparent fashion by enabling a master node to authenticate each device belonging to a group and recording all the message exchanges between authorized participant entities in the blockchain.

Consequences:

Benefits:

- *Integrity:* The creation of trusted zones ensures device integrity while all devices out of a zone are considered malicious and data from them is not pushed to the blockchain.
- *Identity management:* The identification of IoT devices facilitates the implementation of access control policies and authentication mechanisms to ensure security of IoT systems supported by blockchain.

Drawbacks:

- *Compromised trusted authority:* A trusted authority is required to authenticate and identify IoT devices as blockchain nodes, but it could become a bottleneck and a single-point-of-failure in the network.

Related tactic: N/A

Example: This tactic has been identified in Bubbles of Trust [PS79] that creates virtual zones to enable secure communication among devices and consider non-member devices as malicious. This approach requires a master entity acting as a certification authority for enabling followers (i.e., IoT devices) to participate in the virtual zone and sends transactions to the main blockchain for creating a zone at the blockchain level.

5.5 Off-chain Data Storage

Summary: Use offline data storage for recording IoT raw data while keeping a digital hash of the data on-chain for verification.

Motivation: Due to the growth in the number of IoT devices, a huge amount of data is generated in real or near real-time that needs to be analyzed and securely stored to protect it against cyber-attacks [4]. However, the blockchain has limited computation and data storage space, which

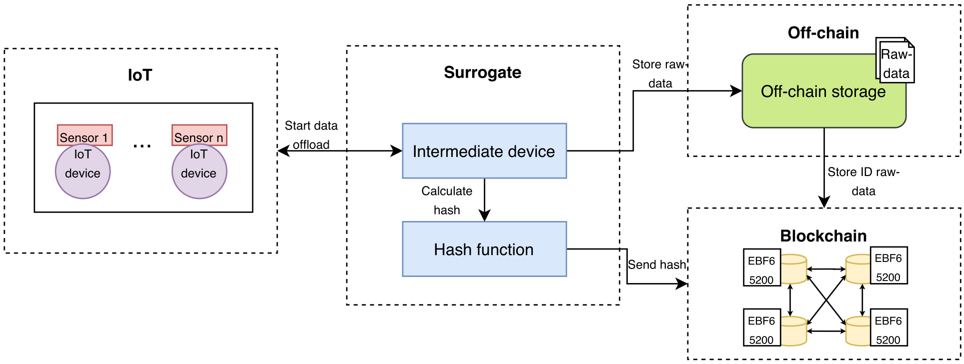

Fig. 6. Off-chain data storage where the surrogates manages IoT raw-data and calculates its hash.

restricts the number of transactions to be recorded on-chain [73]. In addition, the use of public blockchains costs money and could even be more expensive than traditional storage solutions. For instance, Ethereum manages a block gas limit to determine the number, computational complexity, and data size of the transactions included in a block [42].

Solution: Figure 6 shows the main components of the off-chain data storage tactic. It requires a surrogate located in single-hop proximity of IoT devices, offline storage (i.e., local database, private cloud, or P2P storage system), and a blockchain. The basic functioning of this tactic is described as follows [40, 71]. The data generated by IoT devices are sent to a surrogate that operates as an intermediary device between the IoT devices and the blockchain. The surrogate processes IoT data and decides on what data needs to be recorded on-chain or off-chain. Specifically, the IoT raw data are recorded in the off-chain storage while the hash of critical IoT data and the identifier of the raw data is kept on-chain to verify its integrity and immutability. The use of a hash allows to keep a representation of IoT data with a smaller size stored on-chain.

Consequences:

Benefits:

- *Integrity:* To check the integrity of the off-chain data, it is possible to compare the hash of IoT data stored on-chain with the one generated from raw data recorded off-chain.
- *Cost:* Since the use of on-chain storage has a high cost, all the transactions sent to the blockchain can be summarized in a hash to reduce this cost.
- *Data immutability:* Since the identifier of raw data is stored on-chain, any change on the off-chain data can be detected if the interested parties have access to the off-chain data.

Constraints:

- *Privacy:* The blockchain cannot ensure data privacy by design, which increases user concerns about data manipulation and loss of information.
- *Data loss:* Since raw data are stored off-chain, it could be deleted, altered, or manipulated by service providers in the cloud and only its identifier could remain immutable on-chain.
- *Data sharing:* While on-chain data can be securely shared among authorized blockchain nodes via smart contracts, the off-chain data require new approaches for data management.

Related tactic: Encryption data (Section 5.5).

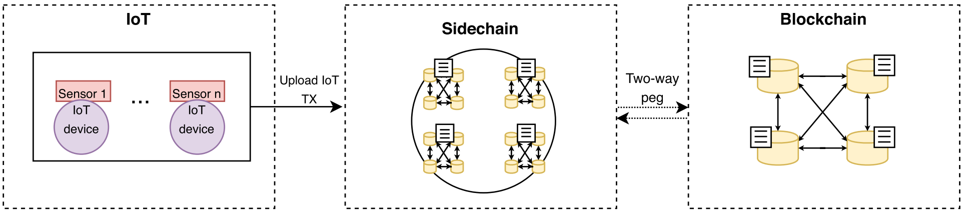

Fig. 7. Sidechain connected to the main blockchain.

Examples:

- *Blockchain auditable storage* [PS8]. The raw data collected by IoT devices are stored off-chain and only its identifier (i.e., hash pointer) is recorded on-chain to ensure its integrity and confidentiality.
- *Blockchain for data sharing* [PS9]. The system summarizes a set of all transactions to be recorded in the blockchain in a digital fingerprint (i.e., hash), which ensures data integrity and transparency. If the integrity of the off-chain data needs to be verified, then a hash of the raw data located off-chain can be generated and compared with the hash of the on-chain data.
- *IoT protection-blockchain* [PS14]. A hash of the raw data is generated and stored on-chain to reduce the cost of on-chain data storage in public blockchains. In addition, this hash of the on-chain transaction is recorded in a local database to verify the integrity of raw data.
- *IoT data assurance* [PS18]. A set of IoT records are compressed using a hash function to get a unique identifier (i.e., hash), which is stored on-chain and off-chain (i.e., cloud solution) to enhance transparency and auditability of IoT data.
- *IoT exchange* [PS20]. The systems distinguish between two types of data: device data and exchange data. The former can be stored in a local database, cloud database, or even a wireless sensor network managed by the owner while the latter is used to keep a track of the data exchange process. In particular, a hash is generated from the IoT raw data and recorded on-chain to verify its integrity.
- *IoT privacy* [PS80]. A decentralized access control model with privacy built-in is proposed where an InterPlanetary File System (IPFS) server is used to group and replicate IoT data in the P2P network without the need for a third party. The hashes of the IPFS files are recorded on-chain through smart contracts and the access control permissions for on-chain data are stored off-chain.

5.6 Sidechain

Summary: Improve the scalability of blockchain by relying on a chain chain that is attached to the main chain using a two-way peg.

Motivation: Due to the increasing amount of data generated by IoT devices, extensive computation and large storage space is required to process and record IoT data securely. However, the blockchain still has limited computation and data storage resources, which puts some restrictions on the adoption of blockchain in IoT systems.

Description: Figure 7 shows the main components of this tactic. A sidechain consists of a blockchain (childchain) attached to a parent chain (main chain) using a two-way peg. This two-way peg mechanism facilitates the use of tokens and assets in another blockchain and then to be moved back to the original blockchain if required [16]. The basic functioning of this tactic is as follows [48]. IoT devices communicate with the sidechain, which in turn regularly commit certain states to the main blockchain to finalize state transitions and secure the system. Each sidechain has its own miners that validate transactions and only periodically report back to the main chain to update its status. This removes bottlenecks on the main chain and increases the speed and scalability of the whole network.

Consequences:

Benefits:

- *Scalability:* The use of sidechain improves the scalability of the main blockchain, since IoT transactions are computed by surrogate chains.
- *Interoperability:* Each IoT application can run in a sidechain and securely exchange digital assets with other surrogate chains at a predetermined rate based on the IoT application requirement.
- *Security:* Each sidechain defines its own level of security and consensus protocol. If a participant in a surrogate chain acts maliciously, then the transactions in the other surrogate chain or in the main blockchain cannot be compromised.

Drawbacks:

- *Cost:* The sidechain has an initial cost, since they need to have enough power for mining and ensuring the safety of IoT transactions.

Related tactic: Two-layer blockchain architecture (Section 5.12).

Example:

- *Optimized blockchain* [PS2]. A modular consortium architecture for IoT and blockchain is proposed where each chain is responsible for processing its own transactions and all the sidechains are attached to the main blockchain. This main chain manages access control permissions and ensures that only authorized users can have access to IoT data from one chain to another chain.
- *Controlchain* [PS67]. A secure architecture is presented to establish relationship attributes and access control authorization between users and devices. The blockchain database is divided into four chains: context, relationship, rules, and accountability, where all of them are attached to the main chain.

Variations: Plasma relies on smart contracts and Merkle Tree to arrange a hierarchical structure where numerous surrogate chains that can communicate and exchange digital assets with the main blockchain [49]. Specifically, plasma implements a treelike structure that consists of child chains, parent chains, and the root chain. Overall, the plasma tactic works as follows [85]. IoT data are sent as transactions to the surrogate chains under the control of the main blockchain. If there are transactions that require a large amount of computational power, then they are continuously broadcast to the main blockchain for validation. As surrogate chains are created from smart contracts, they work independently of each other and handle transactions by defining their own consensus protocol and security rules. Thus, each surrogate chain monitors its own transactions, which eliminates the need for every blockchain node to verify all transactions performed over time in the network [42]. In addition, transactions are moved from the surrogate chains to the main blockchain when it is proven that a participant in a surrogate chain has acted maliciously.

5.7 IoT Devices as Lite Blockchain Nodes

Summary: Connect resource constraint IoT devices to the blockchain network through a gateway device.

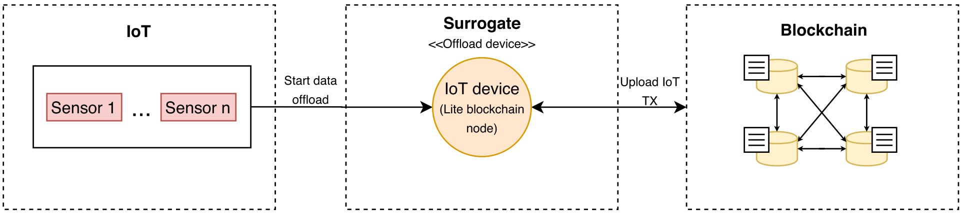

Fig. 8. IoT device as lite blockchain node where the surrogate upload IoT transactions to the blockchain.

Motivation: IoT mainly comprises resource constraint devices with limited computation, storage, and power that do not allow them to implement complex security protocols or process a large amount of data generated by IoT devices [42]. Thus, IoT systems can leverage blockchain technology to record sensor data as immutable and tamper-proof transactions. However, one of the main limitations of blockchain is the limited computation and data storage space, since all transactions are replicated across the blockchain network [30].

Solution: Figure 8 shows the main components of this tactic. To facilitate the adoption of blockchain, the Bitcoin protocol identifies two types of blockchain nodes (i.e., full and lightweight nodes) [42]. The former has enough processing power and storage capacity to process transactions, mine blocks, and keep a full copy of the ledger while the latter can only store their own addresses and send transactions to the full nodes. In IoT systems, devices with high computation capabilities (i.e., Raspberry Pi) can operate as full nodes while the resource-constrained devices (i.e., Arduino) can act as lightweight nodes. The lightweight nodes can send transactions to the full nodes that act as gateways in the network and connect directly to the blockchain [50]. To this end, the full node connects to the blockchain network through a Web3 provider for pushing the received data from resource-constrained devices to the blockchain through a smart contract [52].

Consequence:

Benefits:

- *Low latency:* The data offload operation in the intermediary server decreases latency, since it is located in single-hop proximity to IoT devices.
- *Network efficiency:* The use of Wi-Fi or short-range radio instead of broadband wireless to communicate sensors and the intermediary server reduces bandwidth consumption and improves the user experience.
- *Security:* Critical and sensitive IoT data can be processed and analyzed locally within the IoT network, which could result in better control of the levels of security and privacy.

Drawbacks:

- *Scalabilty:* The integration of resource-constrained and IoT devices with high capabilities improves the scalability of the blockchain network while ensuring the performance [53].
- *Security:* The use of computational powerful IoT devices as gateways increases security and privacy concerns about data manipulation and loss of information [42].

Related tactic: IoT device as full blockchain node (Section 5.8).

- *BIas* [PS23]. The system categorizes IoT devices as full, lightweight, and non-blockchain nodes according to their computation capacity and connection lifetime. In particular, the full nodes transmit the transactions from the lightweight nodes to the blockchain. The non-blockchain nodes are devices with limited capacity that cannot act as full or lightweight client and must connect to a trusted remote node.

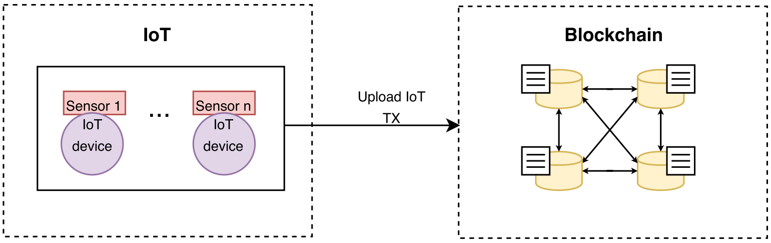

Fig. 9. IoT devices act as full blockchain nodes.

- *Scalable blockchain for IoT* [PS26]. The systems distinguish full, lightweight, and coordination nodes based on their power supply and hardware configuration. The full and lightweight nodes maintain constant and dynamic links in the network respectively and operate as nodes of the blockchain network. The coordination nodes connect devices with dynamic connections to the blockchain network.
- *A two-layer consensus* [PS40]. The system classifies IoT devices into three groups: A (server and backend), B (edge devices, gateways), and C (end devices with low bandwidth). The devices in group A are responsible for connecting devices in group C by transmitting their transactions to the blockchain network while devices in group B can maintain a direct connection.
- *Hybrid-IoT* [PS43]. This system categorizes IoT devices as full, lightweight, and outsider nodes where full nodes participate in the consensus, and mines blocks and lightweight nodes connect to a full node to send transactions to the blockchain. Due to their limited hardware resources, the outsider nodes only sense the environment and their data are not stored in the blockchain to prevent data overload.
- *Blockchain lightweight IoT Clients* [PS30]. IoT devices act as lightweight clients, which only stored their own blockchain addresses and send transactions to the full nodes (i.e., base station). The full nodes consist of a set of wireless base stations that collect transactions from lightweight nodes.
- *Plasma* [PS32]. Plasma enables low-powered IoT devices to operate as lightweight nodes and communicate with edge gateways that act as full nodes. The full nodes view the lightweight nodes as their clients and collect their transactions to send them to the blockchain. Plasma allows low-powered IoT devices to operate as lightweight nodes and communicate with edge gateways that act as full nodes. The full nodes view the lightweight nodes as their clients and collect their transactions to send them to the blockchain.

5.8 IoT Devices as full Blockchain Nodes

Summary: Use IoT devices with high computation capabilities as full nodes to connect directly to the blockchain network.

Motivation: IoT devices with high computation capabilities connect directly to the nearest blockchain node and push transactions to the blockchain network. A full blockchain node keeps a copy of the complete blockchain and validates its own transactions as well as other transactions in the blockchain network.

Solution: Figure 9 shows the main components of this tactic. The powerful IoT devices like Raspberry Pi can be connected directly to the blockchain and operates as a full blockchain node [44]. This device acts as a connector that provides communication channels and local services to resource-constrained IoT devices. Specifically, the connector takes the role of full nodes by processing transactions and participating in the consensus protocol. To this end, the connector communicates with the nearest blockchain node through a Web3 provider and uploads IoT data to the blockchain network via smart contract [42].

Consequences:

Benefits:

- *Latency efficiency:* The deployment of a connector located in single-hop proximity of IoT devices for data processing and connecting directly to the blockchain minimizes latency in the network.
- *Bandwidth reduction:* The connector deployed at the edge of the network proximate to IoT devices results in less demand for bandwidth, since data are processed locally instead of sending it to the cloud.

Constraints:

- *Lack of confidentiality:* The connector could raise security and privacy concerns about data manipulation and loss of information, since all the data are available in the connector and could be manipulated and altered by malicious users.
- *Single-point-of-failure:* The deployment of a single server for processing and storing large volumes of data could become a single point of failure and bottleneck in the network as the amount of IoT transactions increase over time.
- *Cost:* The integration of devices with strong computation capabilities to connect directly to the blockchain could increase the implementation and maintenance costs.

Related tactic: IoT devices as lite blockchain node (Section 5.7).

Examples:

- *Optimized blockchain* [PS2]. IoT devices with high computational resources can act as gateways in the network and transmit IoT data to the blockchain.
- *Blockchain Meets IoT* [PS75]. The system uses a management hub to connect IoT devices to Ethereum nodes through RPC calls and a JavaScript library.
- *IoT protection-blockchain* [PS14]. The system introduces intermediary servers between IoT devices and the blockchain to perform real-time processing tasks before transmitting the results to the blockchain network. Here, a publish-subscribe mechanism is used for handling computation and power-consuming resources in the blockchain network.
- *IoT data assurance* [PS18]. The control system aggregate the collected data from drones and calculate its hash before recording it to the blockchain and cloud to ensure its integrity.

5.9 Caching Offload

Summary: Use a cache system to offload a subset of IoT transactions processed by blockchain to make faster data operation requests.

Motivation: Due to the constraint resources in the majority of IoT devices, IoT systems mainly leverage on computational and storage capabilities in the cloud [52]. However, access to sensor data in the cloud demands over a multi-hop proximity and lower bandwidth connection that increases latency in IoT transactions and bandwidth consumption.

Description: Figure 10 illustrates the main components of this tactic. The caching offload tactic requires sensors running on the IoT layer, an intermediary server operating as a surrogate and a shared data storage running on the blockchain. The basic functioning of this tactic is described as follows [69]. The sensor data coming from thousands of IoT devices are sent to the intermediary

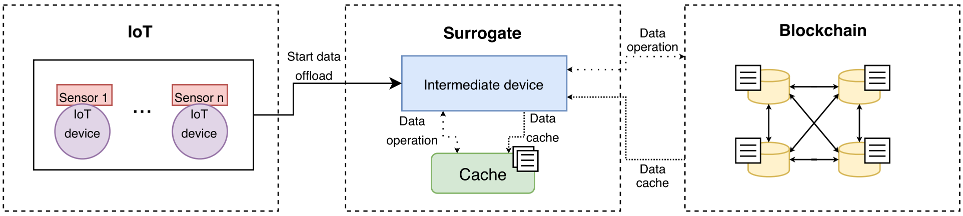

Fig. 10. Caching offload where the surrogate manages a cache system.

server for processing and analysis before recording it as transactions in the blockchain. To make faster data operations, the surrogate retrieves data from the blockchain and stores it locally so it is available to IoT devices when they need it. Therefore, access to the blockchain is only necessary when data are not available on the surrogate, which minimizes latency in the network.

Consequences:

Benefits:

- *Improved latency:* Using caching, data retrieval from edge servers could be faster compared to the retrieving all the information from the blockchain. This improvement in the data access also has an impact on the overall performance of the system.
- *Low throughput:* The use of caching improves transaction throughput in the blockchain network, because the edge server enables faster access to on-chain data.

Drawbacks:

- *Interoperability:* The interoperation between the caching system running on the surrogate and the blockchain nodes could be difficult and increase security concerns about data management.
- *Stale data:* The use of a cache system could lead to stale data where in each data request previously recorded data can be fetched instead of new value of the data.

Related tactic: N/A

Example: [PS66] suggests an example of application of the caching offloading tactic in Edge and Caching. Due to the limited capabilities of IoT devices, the blockchain nodes rely on caching system implemented on edge servers for reaching consensus and caching resources. It ensures fast data access and improves the performance of the system.

Variation: A hardware-based caching [PS92]. This system proposes a cache technique using a Field Programmable Gate Array-based (FPGA) Network Interface Card (NIC) to process data requests from IoT devices before sending them to the blockchain. In particular, data requests are handled in the FPGA internal memory and storage and pushed to the blockchain as transactions, which reduces latency and bandwidth in the network and improves transaction confirmation.

5.10 Surrogate Computation

Summary: Offload computation-intensive blockchain tasks (i.e., transaction processing and keeping a copy of the complete blockchain) to a surrogate to reduce computation and data storage on-chain.

Motivation: In public blockchains, the miner and full nodes are required to store the complete copy of the ledger and validate every transaction in order [42]. This feature enhances the security of IoT systems but can also overload blockchain nodes with computation and data storage requirements due to the large amount of data sent as transactions to the blockchain [43]. In

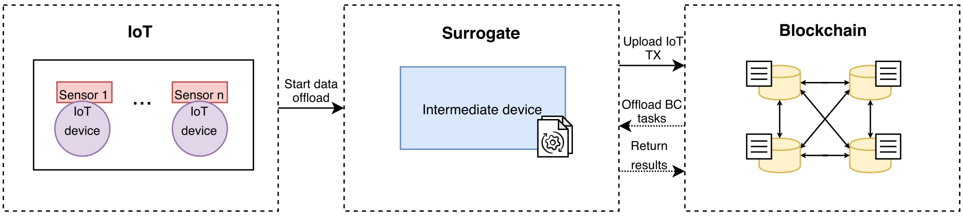

Fig. 11. Surrogate computation where the surrogate processes blockchain tasks.

addition, the requirement of keeping a complete copy of IoT transactions in blockchain nodes limits the integration of IoT devices as full blockchain nodes due to their constraint resources.

Description: Figure 11 shows the main component of this tactic. The surrogate computation tactic requires a cloud server and a share data storage running in the cloud and a blockchain, respectively. This pair of components communicate to coordinate computation-intensive tasks. The basic functioning of this tactic is as follows. First, the sensor data coming from IoT devices need to be uploaded as transactions to a blockchain to enhance its immutability and integrity. Due to the limited computation and data storage space in a public blockchain, it establishes connectivity to a server in the cloud for offloading tasks that require extensive computation (i.e., hash calculation and transaction processing) [18]. Once the task is completed and the results of data operation are sent back to a blockchain for verification [69]. If the hash is correct, then a blockchain node generates a new block and broadcasts it to all nodes in the P2P network. Each node receives the new block and validates it in consensus before adding it to the end of the chain.

Consequences:

Benefits:

- *Computation efficiency:* The use of a cloud server as a surrogate helps to reduce the computation and data storage loads in blockchain nodes and reduce latency in transaction confirmation.

Drawbacks:

- *Data immutability:* Since blockchain connects to a cloud server for processing data and executing computation-intensive tasks, IoT data could be altered and manipulated by service providers in the cloud.

Related tactic: N/A

Example: The systems that implement the surrogate tactic maintain a list of the edge servers who are allowed to connect to the blockchain and call smart contract functions. For instance, Edge and Caching [PS66] relies on edge servers to maintain a P2P network and execute computational expensive tasks like hashing. Specifically, the authorized edge servers in the smart contract receive all the required information to calculate the hash and return the output to a blockchain network for verification. This verification process consists of a proof of execution on-chain.

5.11 Sharding

Summary: The majority of IoT systems have real-time data sharing requirements that demand improvements of transaction confirmation time in the blockchain. The use of sharding increases transactional throughput in blockchain networks with minimal disruption to the IoT users.

Motivation: The large amount of data generated by IoT devices results in a high number of transactions to be uploaded in a blockchain. However, in a public blockchain, miner nodes are

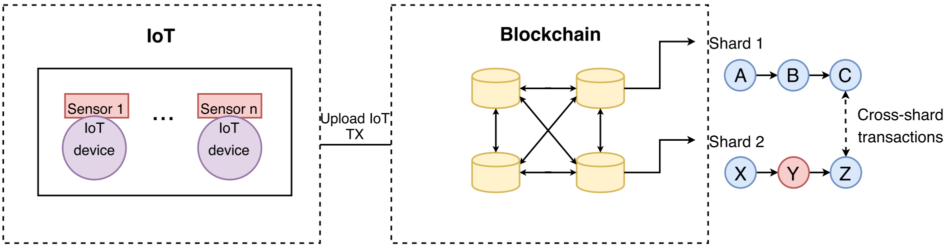

Fig. 12. Sharding where the surrogate handles shards.

responsible for keeping a complete copy of the ledger and validating every transaction in order. Since the blockchain cannot process more transactions than the capacity of a single node, it could become a bottleneck in case of a high number of transactions.

Description: Figure 12 shows the main components of this tactic. The sharding tactic consists of spreading out the computation and data load across the blockchain network to reduce transaction confirmation time [42]. It means that a subset of miner nodes process a subset of transactions generated by IoT devices instead of processing the entire transactions in the blockchain network. Each node is only responsible for keeping information related to its partition (i.e., shard) and maintaining its own transaction history [41]. The subset of the transaction consists of a header (i.e., identifier) and body (i.e., all transactions belonging to a specific group). Once a transaction is verified within a shard, the entire shard is updated to ensure that all nodes within the shard have the same information. In addition, a shard can trigger events to other shards for exchanging digital assets, which is known as cross-shard communication. These arrangements ensure that multiple transactions can be processed simultaneously and enhance security in the blockchain network.

Consequences:

Benefits:

- *Faster transactions:* Since one of the main advantages of sharding is to process transactions in parallel, it can process 10 times of the number of transactions performed by traditional blockchains per second.
- *Low data storage and cost:* The use of shards facilitates storing a large amount of IoT data as transactions at low cost, because each blockchain node handles a small portion of the data or keeping the complete copy of the ledger.

Drawbacks:

- *Data sharing:* The sharding enables transaction exchanges across shards but the cross-shard communication is still challenging. When a specific participant in one shard requires information that is not within its shard, it has to identify which shards contain the required information and exchange it for transaction processing.

Related tactic: N/A

Example: An example of the sharding tactic has been identified in [PS42]. The system consists of multiple micro-blockchains also known as shards, where each of them is responsible for receiving transactions, broadcasting them to the P2P network, and making the consensus. Each shard and the main shard run the PBFT consensus protocol twice to reach a consensus and create the blocks. The main shard consists of some important nodes in each shard that are responsible for making the final consensus and generating the blocks to be attached to the main chain.

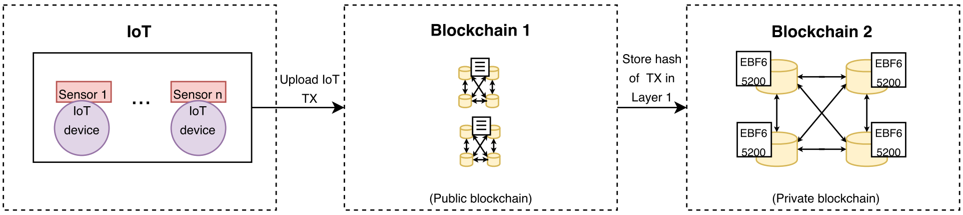

Fig. 13. Two-layer consensus that supports a public and private blockchain.

5.12 Two-layer Blockchain Architecture

Summary: Enhance the scalability of blockchain by relying on a two-layer blockchain architecture.

Motivation: With thousands of sensors collecting data from the environment, a huge number of transactions need to be uploaded to the blockchain [42]. Since sensor data are replicated in all blockchain nodes, it affects the blockchain size and influences the consensus protocol. In addition, the high storage requirements for blockchain systems put more limitations on the integration of resource constraint IoT devices as blockchain nodes [52].

Description: Figure 13 shows the main components of this tactic. The two-layer consensus tactic consists of multiple public blockchains and a surrogate chain running on the first and the second layer respectively that coordinate the on-chain data operation. The basic functioning of this tactic is described as follows [53]. The first layer comprises multiple public blockchains where each of them calculates a hash from a group of transactions and periodically sends it to the second layer. If a malicious user changes any transaction in the first layer, then it could be detected as the hash of transactions recorded in the first layer is sent to the second layer. The second layer can operate as a surrogate running a private blockchain that verifies the correctness, completeness, and authenticity of transactions in the first layer. For instance, PoW and PBFT can be used as consensus protocols in the first and second layers, respectively, to achieve interoperability, improve scalability, and reduce transaction confirmation time. In addition, every miner in the first layer can be associated with offline data storage to reduce the number of transactions to be sent to the second layer.

Consequences:

Benefits:

- *Scalability:* The use of multiple public blockchains and a private blockchain improves the scalability of the blockchain network, since IoT transactions are logged into a distributed database. In addition, the use of offline data storage can alleviate the storage requirements in the blockchain nodes in the second layer.
- *Integrity:* Since the hash of IoT transactions processed in the first layer is periodically recorded in the second layer, it makes it easy to detect data forgery.

Drawbacks:

- *Limitation in private blockchain:* Using a private blockchain, the maximum number of validators tested in previous works is 20 [42], which makes difficult the integration of thousands of powerful IoT devices as blockchain nodes.

Related tactic: Off-chain data storage (Section 5.5).

Example:

- *Hybrid-IoT* [PS43]. A sub-blockchain based on PoW is created to achieve distributed consensus among IoT devices as nodes of the blockchain network. Each sub-blockchain consists of a group of IoT devices that follows a set of rules called sweet-spot guidelines to define in which way IoT devices can establish a sub-blockchain. All sub-blockchains are connected to a PBFT interconnector framework that handles the interoperability among multiple sub-blockchains.
- *Two-layer consensus* [PS40]. The base-layer and top-layer are deployed and different class nodes are defined based on their capabilities and connection lifetime. On one hand, the base-layer consists of class B and C nodes and considers a hybrid consensus protocol (i.e., PoW and PoS) to improve the scalability and transaction time of the blockchain network. On the other hand, the top-layer is formed by class A nodes selected by managers and ensures base-layer blocks are performed in a randomly and transparent manner by following a non-byzantine fault tolerance algorithm.

6 FINDINGS AND GAPS FROM PRIMARY STUDIES

In this section, we summarize the most noticeable observations and present some gaps and opportunities for architecting IoT systems supported by blockchain. In particular, we carefully position our discussion in light of the integration of these two technologies from the software architecture perspective.

- *Lack of architectural support for some quality attributes.* Section 4 presents the most commonly reported quality attributes in the literature for the architectural design of IoT systems supported by blockchain. In addition to security, performance, and scalability, our analysis reveals that there are other quality attributes such as interoperability, efficiency, adaptability, and mobility that can be keys to reason about the dynamism and uncertainties in IoT systems supported by blockchain. However, these quality attributes are only briefly mentioned in the primary studies and lack architectural support in the reviewed literature. For instance, IoT devices can join and leave the network at any time, which could make it easy for attackers to compromise such devices with fake identifiers and to manipulate IoT networks in the presence of such dynamic networks [50]. From the blockchain perspective, the business logic encoded on smart contracts variables could require updating based on the IoT context [40]. Thus, we highlight the need for research and development on providing architectural support for interoperability, efficiency, adaptability, and mobility in IoT systems supported by blockchain.
- *Lack of focus on the integration of blockchain and IoT from the software architecture perspective.* The systems in the primary studies tend to have a little discussion of the design decisions to consider when architecting IoT systems supported by blockchain. Most of them are designed in an ad hoc manner, and lack systematic analysis of design alternatives and their impacts on the quality attributes for this category of systems. Only a few studies agree on allocation of data, and computation on-chain or off-chain is a major decision when architecting IoT systems supported by blockchain [37, 38, 40, 73]. This observation shows that architectural tactics for the integration of blockchain and IoT is still an area for exploration that could greatly benefit software developers and architects.
- *Lack of focus on the system-level concerns.* The majority of primary studies tend to have a narrow focus on proving that powerful IoT devices can be effectively integrated to blockchain due to the limited capabilities and connection lifetime of the majority of IoT devices. There are questions related to the integration of both technologies that need to be addressed when

systems grow from initial prototypes to operational systems with hundreds of IoT devices, such as:

- How do the systems perform when the blockchain network is hosted in the edge with IoT devices trying to transmit the collected data to the same edge node for pre-processing and blockchain tasks?
- In the same scenario, what happens when IoT devices lose connectivity to the blockchain network running on edge nodes?
- How can IoT devices know that the blockchain nodes running on the edge layer are trustworthy, to send transactions to them?
- In those systems that deploy a blockchain network based on cloud resources, what are the mechanisms for ensuring that IoT data are protected when it is in transit?
- What are the tradeoffs between quality attributes promoted by blockchain design configuration and other quality attributes (i.e., network usage, energy efficiency, and latency) that can impact the design of blockchain-based IoT systems?

- *Lack of large-scale evaluation.* Many systems in the studies use a Proof-of-Concept to demonstrate the feasibility of integrating IoT systems with blockchain, which are implemented on a blockchain testnet or local environments [39]. For instance, the consensus protocol in the Ethereum testnet (i.e., Kovan and Rinkeby) is PoA instead of PoW that is the de facto consensus protocol in public blockchains. Thus, the results shown in the evaluation section of the primary studies are inaccurate and differ in terms of latency from the public Ethereum blockchain. In addition, the experiments are run on controlled environments over Wi-Fi connections and with a small number of IoT devices, which could not reflect real IoT systems with thousands of heterogeneous devices collecting real-time data and transmitting it to the blockchain network.
- *Lack of focus on the main issues of blockchain for IoT systems.* Most of the studies in the reviewed literature show that there is a lack of focus on the IoT-based consensus protocol, transaction validation rules, and secure device integration. Therefore, research is required to create IoT oriented consensus protocols that minimizes latency and energy while ensuring the security and privacy of IoT systems. In addition, due to the limited memory and storage of IoT devices, they cannot store the increasing amounts of data uploaded to blockchain nodes. Hence, to improve the scalability of blockchain and enhance the integration of blockchain and IoT, IoT systems could take advantage of fog nodes that can preprocess data before sending them to the blockchain. However, there is a lack of IoT-centric transaction validation rules where IoT transactions can be validated rapidly without causing bottlenecks in the network. A possible solution is the use of off-chain storage for processing IoT transactions instead to wait for block confirmation.

7 THREATS TO VALIDITY

In this section, we summarize the identified threats to validity in our study and how we deal with them.

External validity: Among the potential external threats in our study, we highlight the fact of having a limited set of primary studies, which might not represent the state-of-the-art and practices on architecting IoT systems supported by blockchain. To mitigate this threat, we applied a search strategy on the selected primary studies following the guidelines suggested by Kitchenham and Brereton [32], which was combined with a snowballing technique to enlarge the set of studies collected from the automatic search. We only included peer-reviewed studies (i.e., journals, conferences, and workshops) and excluded non-scientific studies (i.e., blogs, tutorials, etc.) as they do not reliability deliver high-quality scientific contributions. We also defined the inclusion and exclusion criteria, which were revised and refined by researchers and experts in the field. Specifically, we discussed the definition of each inclusion and exclusion criterion to have a minimal bias on the identification of those primary studies and provide direct evidence about the proposed research questions. It is important to highlight that even when we defined E2 for limiting secondary studies (i.e., surveys and systematic reviews), we considered them for assessing the completeness of our set of selected studies and for identifying significant challenges in architecting IoT systems supported by blockchain.

Internal validity: We limited the level of influence of extraneous variables in our study by defining a rigorous research protocol, which was developed in consultation with the co-authors and with other researchers. This protocol describes each stage of the conducted study, including (i) string search derived from the research questions, (ii) the selection criteria to identify relevant studies, and (iii) the data analysis to extract relevant information from the set of final primary studies.

Construct validity: We performed an automatic search on the largest databases and indexing libraries in computer science and software engineering to collect our primary studies [32, 47]. We also defined a search string using the terms derived from the research questions and their synonyms to identify as many studies as possible to extend the coverage of the automatic search. In addition, we designed a rigorous and explicit set of inclusion and exclusion criteria to identify primary studies that have direct evidence of the research questions. We ensured the validity of the collected primary studies by performing an automatic search on multiple well-known scientific databases and indexing libraries in computer science and software engineering [32, 47]. We did not restrict our search of primary studies to publication date to extend the coverage of the automatic search. As some studies lack architectural definition, we performed a title-, abstract-, and full-text reading to reduce misinterpretation in the selection process.

Conclusion validity: We mitigated the potential threats regarding the relationship between the extracted data and obtained results by applying a well-defined and rigorous search protocol that was defined following the most recent guidelines on systematic mapping studies [32, 47]. We also revised and refined the protocol with experts in the field to ensure its completeness and applicability. This work applied qualitative and quantitative analysis to describe the results of our study in terms of the proposed research questions and used the extracted data for further analysis (i.e., quality tradeoffs, constraints, and dependencies among tactics, etc.). We documented each stage of our study to facilitate its understanding and replication by independent researchers.

8 CONCLUSION AND FUTURE WORK

In this work, we have studied the state-of-the-art and practices in the integration of blockchain and IoT from the software architecture perspective. To address the general lack of architectural support for the design of IoT systems supported by blockchain, we provided a catalog of architectural tactics for the design of this category of systems and explained how they could influence the achievement in the quality attributes of interest. Specifically, an SLR was conducted to investigate the commonly reported quality attributes and design decisions that need to be considered when architecting IoT systems supported by blockchain. Our study aims to provide empirical evidence on the architectural design of IoT systems supported by blockchain and to identify gaps in the current literature for future research and development.

From the primary studies, we identified security, scalability, and performance as the commonly reported quality attributes. However, there are other concerns such as adaptability, mobility, and reliability that also need to be considered when architecting this category of systems. We also extracted twelve tactics from the reviewed literature for supporting the design of IoT systems supported by blockchain. We can not claim that our architectural tactics are exhaustive; instead, out goal was to cover as many primary studies as possible to identify the tactics commonly reported by architects and practitioners.

Our findings have also identified new areas for future research: (i) despite the significance of the identified quality attributes, there are other requirements that lack architectural support in the literature; (ii) investigation is required to evaluate the real-world impact of the architectural tactics in this category of systems; and (iii) additional research is needed to explore the tradeoffs among the quality attributes and identified tactics. These opportunities for research require intensive collaboration between academia and industry considering the fact that meaningful IoT systems consist of thousands of devices, which collect a large amounts of data that need to be processed and analyzed. It is worth noting that this study can guide software architects to more rigorously design this category of systems by concretizing the body of knowledge on architecting IoT systems supported by blockchain. To shed more light on the integration of blockchain and IoT, in our future work we aim to implement an experimental testbed to evaluate the identified architectural tactics in terms of their most important tradeoffs and dependencies.

APPENDIX

A LIST OF PRIMARY STUDIES

Table 8 presents the list of the 100 primary studies.

Table 8. Primary Studies

ID	Title	Short name	Author(s)	Year
PS1	Towards blockchain-based intelligent transportation systems	Blockchain Transportation	Yuan, Yong and Wang, Fei-Yue	2016
PS2	Towards an optimized blockchain for IoT	Optimized blockchain	Dorri, Ali and Kanhere, Salil S and Jurdak, Raja	2017
PS3	Block-VN: A Distributed Blockchain Based Vehicular Network Architecture in Smart City	Block-VN	Sharma, Pradip Kumar and Moon, Seo Yeon and Park, Jong Hyuk	2017
PS4	MeDShare: Trust-Less Medical Data Sharing among Cloud Service Providers via Blockchain Using Blockchains to Strengthen the Security of IoT	MeDShare	Xia, QI and Sifah, Emmanuel Boateng and Asamoah, Kwame Omono and Gao, Jianbin and Du, Xiaojiang and Guizani, Mohsen	2017
PS5	A Software Defined Fog Node Based Distributed Blockchain Cloud Architecture for IoT	SDN-Blockchain	Sharma, Pradip Kumar and Chen, Mu-Yen and Park, Jong Hyuk	2017
PS6	Blockchain Based Distributed Control System for Edge Computing	Blockchain for Edge	Stanciu, Alexandru	2017
PS7	Towards better availability and accountability for IoT updates by means of a blockchain	IoT Updates	Boudguiga, Aymen and Bouzerna, Nabil and Granboulan, Louis and Olivereau, Alexis and Quesnel, Flavien and Roger, Anthony and Sirdey, Renaud	2017
PS8	Towards blockchain-based auditable storage and sharing of IoT data	Blockchain auditable storage	Shafagh, Hossein and Burkhalter, Lukas and Hithnawi, Anwar and Duquennoy, Simon	2017
PS9	Integrating blockchain for data sharing and collaboration in mobile healthcare applications Blockchain for data sharing	Blockchain for data sharing	Liang, Xueping and Zhao, Juan and Shetty, Sachin and Liu, Jihong and Li, Danyi	2018
PS10	Peer to peer for privacy and decentralization in the internet of things	P2P privacy in IoT	Conoscenti, Marco and Vetro, Antonio and De Martin, Juan Carlos	2017

Table 8. Primary Studies

ID	Title	Short name	Author(s)	Year
PS11	Vegvisir: A Partition-Tolerant Blockchain for the Internet-of-Things	Vegvisir	Karlsson, Kolbeinn and Jiang, Weitao and Wicker, Stephen and Adams, Danny and Ma, Edwin and van Renesse, Robbert and Weatherspoon, Hakim	2018
PS12	Blockchain based hybrid network architecture for the smart city	Hybrid BC-IoT	Sharma, Pradip Kumar and Park, Jong Hyuk	2018
PS13	An Out-of-band Authentication Scheme for Internet of Things Using Blockchain Technology	IoT authentication	Wu, Longfei and Du, Xiaojiang and Wang, Wei and Lin, Bin	2018
PS14	Towards using blockchain technology for IoT data access protection	IoT protection-blockchain	Rifi, Nabil and Rachkidi, Elie and Agoulmine, Nazim and Taher, Nada Chendeb	2018
PS15	On design issues and architectural styles for blockchain-driven IoT services	Architectural styles	Liao, Chun-Feng and Bao, Sheng-Wen and Cheng, Ching-Ju and Chen, Kung	2017
PS16	IoTChain: A blockchain security architecture for the Internet of Things	IoTChain	Alphand, Olivier and Amoretti, Michele and Claeys, Timothy and Dall'Asta, Simone and Duda, Andrzej and Ferrari, Gianluigi and Rousseau, Franck and Tourancheau, Bernard and Veltri, Luca and Zanichelli, Francesco	2018
PS17	Blockchain as a Service for IoT	Blockchain as a Service for IoT	Samaniego, Mayra and Deters, Ralph	2016
PS18	Towards data assurance and resilience in IoT using blockchain	IoT data assurance	Liang, Xueping and Zhao, Juan and Shetty, Sachin and Li, Danyi	2017
PS19	Blockchain platform for industrial internet of things	BC-IIoT	Bahga, Arshdeep and Madisetti, Vijay K	2016
PS20	A decentralized solution for IoT data trusted exchange based-on blockchain	IoT exchange	Huang, Zhiqing and Su, Xiongye and Zhang, Yanxin and Shi, Changxue and Zhang, Hanchen and Xie, Luyang	2017
PS21	Adaptable blockchain-based systems: A case study for product traceability	Adaptabe blockchain	Lu, Qinghua and Xu, Xiwei	2017
PS22	An Approach to Data Privacy in Smart Home using Blockchain Technology	Privacy SH	Dang, Thanh Long Nhat and Nguyen, Minh Son	2018
PS23	BlAsT: Blockchain-Assisted Key Transparency for Device Authentication	BlAsT	Gattolin, Alessandro and Rottondi, Cristina and Verticale, Giacomo	2018
PS24	IoT data integrity verification for cyber-physical systems using blockchain	Integrity CPS	Machado, Caciano and Fröhlich, Antônio Augusto Medeiros	2018

(Continued)

Table 8. Continued

ID	Title	Short name	Author(s)	Year
PS25	An architecture pattern for trusted orchestration in IoT edge clouds	Pahl, Claus and El Ioini, Nabil and Helmer, Sven and Lee, Brian	2018	
PS26	A dynamic scalable blockchain based communication architecture for IoT	Scalable blockchain for IoT	Qiu, Han and Qiu, Meikang and Memmi, Gerard and Ming, Zhong and Liu, Meiqin	2018
PS27	Approaches to Front-End IoT Application Development for the Ethereum Blockchain	Front-End IoT Dev	Pustišek, Matevž and Kos, Andrej	2018
PS28	A Peer-to-Peer Architecture for Distributed Data Monetization in Fog Computing Scenarios	P2P Data Monetization	de la Vega, Francisco and Soriano, Javier and Jimenez, Miguel and Lizcano, David	2018
PS29	Blockchain-Based Internet of Vehicles: Distributed Network Architecture and Performance Analysis	Blockchain-based IoV	Jiang, Tigang and Fang, Hua and Wang, Honggang	2019
PS30	Delay and Communication Tradeoffs for Blockchain Systems With Lightweight IoT Clients	Blockchain Lightweight IoT clients	Danzi, Pietro and Kalør, Anders E and Stefanović, Čedomir and Popovski, Petar	2019
PS31	BIFF: A Blockchain-based IoT Forensics Framework with Identity Privacy	BIFF	Le, Duc-Phong and Meng, Huasong and Su, Le and Yeo, Sze Ling and Thing, Vrizlynn	2019
PS32	Integration of Fog Computing and Blockchain Technology Using the Plasma Framework	Fog and blockchain using Plasma	Ziegler, Michael Herbert and Groβmann, Marcel and Krieger, Udo R	2019
PS33	Emergency Service for Smart Home System Using Ethereum Blockchain: System and Architecture	Emergency SH	Tantidham, Thitinan and Aung, Yu Nandar	2019
PS34	New Blockchain-Based Architecture for Service Interoperations in Internet of Things	Interoperability IoT	Viriyasitavat, Wattana and Da Xu, Li and Bi, Zhuming and Sapsomboon, Assadaporn	2019
PS35	IoT Meets Blockchain: Parallel Distributed Architecture for Data Storage and Sharing	IoT Meets Blockchain	Liu, Shaowei and Wu, Jing and Long, Chengnian	2018
PS36	An Efficient Forensics Architecture in Software-Defined Networking-IoT Using Blockchain Technology	Forensic SDN	Pourvahab, Mehran and Ekbatanifard, Gholamhossein	2019
PS37	Privacy Improvement Architecture for IoT	Privacy IoT	Addo, Ivor D and Ahamed, Sheikh I and Yau, Stephen S and Buduru, Arun	2018
PS38	Blockchain and IoT Data Analytics for Fine-Grained Transportation Insurance	Blockchain transport insurance	Li, Zengxiang and Xiao, Zhe and Xu, Quanqing and Sotthiwat, Ekanut and Goh, Rick Siow Mong and Liang, Xueping	2018

(Continued)

Table 8. Continued

ID	Title	Short name	Author(s)	Year
PS39	Blockchain-based Ownership Management for Medical IoT (MIoT) Devices	MIoT	Alblooshi, Mansoor and Salah, Khaled and Alhammadi, Y	2019
PS40	A Two-Layer-Consensus Based Blockchain Architecture for IoT	Two-layer consensus	Bai, He and Xia, Geming and Fu, Shaojing	2019
PS41	Maximizing the System Energy Efficiency in the Blockchain Based Internet of Things	Energy blockchain	Fu, Shu and Zhao, Lian and Ling, Xinhua and Zhang, Haijun	2019
PS42	A Hierarchical Sharding Protocol for Multi-Domain IoT Blockchains	Sharding	Tong, Wei and Dong, Xuewen and Shen, Yulong and Jiang, Xiaohong	2019
PS43	Hybrid-iot: Hybrid blockchain architecture for internet of things-pow sub-blockchains	Hybrid-IoT	Sagirlar, Gokhan and Carminati, Barbara and Ferrari, Elena and Sheehan, John D and Ragnoli, Emanuele	2016
PS44	Management and monitoring of IoT devices using blockchain	Management IoT	Košt'ál, Kristián and Helebrandt, Pavol and Belluš, Matej and Ries, Michal and Kotuliak, Ivan	2019
PS45	Fog Computing Architecture Based Blockchain for Industrial IoT	Fog IIoT	Jang, Su-Hwan and Guejong, Jo and Jeong, Jongpil and Sangmin, Bae	2019
PS46	Blockchain-based secure firmware management system in IoT environment	Blockchain firmware IoT	Son, Minsung and Kim, Heeyoul	2019
PS47	Privacy-preserving blockchain based IoT ecosystem using attribute-based encryption	Privacy-preserving blockchain	Rahulamathavan, Yogachandran and Phan, Raphael C-W and Rajarajan, Muttukrishnan and Misra, Sudip and Kondoz, Ahmet	2017
PS48	An Efficient and Compacted DAG-based Blockchain Protocol for Industrial Internet of Things	DAG	Cui, Laizhong and Yang, Shu and Chen, Ziteng and Pan, Yi and Xu, Mingwei and Xu, Ke	2019
PS49	Opportunistic Mobile IoT with Blockchain Based Collaboration	Opportunistic IoT	Chamarajnagar, Ravishankar and Ashok, Ashwin	2018
PS50	MediChainTM: A Secure Decentralized Medical Data Asset Management System	MediChainTM	Rouhani, Sara and Butterworth, Luke and Simmons, Adam D and Humphery, Darryl G and Deters, Ralph	2018
PS51	Homomorphic Consortium Blockchain for Smart Home System Sensitive Data Privacy Preserving	Homomorphic blockchain	She, Wei and Gu, Zhi-Hao and Lyu, Xu-Kang and Liu, Qi and Tian, Zhao and Liu, Wei	2019
PS52	A Blockchain-Based Decentralized Security Architecture for IoT	Decentralized architecture for IoT	Angin, Pelin and Mert, Melih Burak and Mete, Okan and Ramazanli, Azer and Sarica, Kaan and Gungoren, Bora	2018
PS53	Analysis of the Communication Traffic for Blockchain Synchronization of IoT Devices	Traffic for blockchain	Danzi, Pietro and Kalor, Anders Ellersgaard and Stefanovic, Cedomir and Popovski, Petar	2018

(Continued)

Table 8. Continued

ID	Title	Short name	Author(s)	Year
PS54	Using Blockchains to Strengthen the Security of IoT	Strengthen IoT Security	Kouzinopoulos, Charalampos S and Spathoulas, Georgios and Giannoutakis, Konstantinos M and Votis, Konstantinos and Pandey, Pankaj and Tzovaras, Dimitrios and Katsikas, Sokratis K and Collen, Anastasija and Nijdam, Niels A	2018
PS55	Decentralized On-Demand Energy Supply for Blockchain in Internet of Things: A Microgrids Approach	Energy for blockchain	Li, Jianan and Zhou, Zhenyu and Wu, Jun and Li, Jianhua and Mumtaz, Shahid and Lin, Xi and Gacanin, Haris and Alotaibi, Sattam	2019
PS56	Blockchain Based Authentication and Authorization Framework for Remote Collaboration Systems	Blockchain-based Authentication	Widick, Logan and Ranasinghe, Ishan and Dantu, Ram and Jonnada, Srikanth	2019
PS57	Design and implementation of an automated and decentralized pollution monitoring system with blockchains, smart contracts, and LoRaWAN	Design IoT	Niya, Sina Rafati and Jha, Sanjiv S and Bocek, Thomas and Stiller, Burkhard	2018
PS58	Chained of Things: A Secure and Dependable Design of Autonomous Vehicle Services	Chained of Things	Hasan, Md Golam Moula Mehedi and Datta, Amarjit and Rahman, Mohammad Ashiqur and Shahriar, Hossain	2018
PS59	Blockchain and IoT-Based Cognitive Edge Framework for Sharing Economy Services in a Smart City	Blockchain and IoT-Based	Rahman, Md Abdur and Rashid, Md Mamunur and Hossain, M Shamim and Hassanain, Elham and Alhamid, Mohammed F and Guizani, Mohsen	2019
PS60	Enhancing IoT Security and Privacy Using Distributed Ledgers with IOTA and the Tangle	IoT Security and Privacy	Shabandri, Bilal and Maheshwari, Piyush	2019
PS61	A User Authentication Scheme of IoT Devices using Blockchain-Enabled Fog Nodes	Authentication IoT	Almadhoun, Randa and Kadadha, Maha and Alhemeiri, Maya and Alshehhi, Maryam and Salah, Khaled	2018
PS62	Using Blockchain to Support Data and Service Management in IoV/IoT	Blockchain for data	Odiete, Obaro and Lomotey, Richard K and Deters, Ralph	2017
PS63	Blockchain and the Internet of Things: A Software Architecture Perspective	BC-IoT	Liao, Chun-Feng and Hung, Chien-Che and Chen, Kung	2019
PS64	Managing IoT devices using blockchain platform	Managing IoT	Huh, Seyoung and Cho, Sangrae and Kim, Soohyung	2017
PS65	Work-in-progress: Integrating low-power IoT devices to a Blockchain-Based Infrastructure	Work-in-progress	Özyılmaz, Kazım Rıfat and Yurdakul, Arda	2017

(Continued)

Table 8. Continued

ID	Title	Short name	Author(s)	Year
PS66	Edge Computing and Caching based Blockchain IoT Network	Edge and Caching	Xu, Fangmin and Yang, Fan and Zhao, Chenglin and Fang, Chao	2018
PS67	Controlchain: Blockchain as a central enabler for access control authorizations in the IoT	Controlchain	Pinno, Otto Julio Ahlert and Gregio, Andre Ricardo Abed and De Bona, Luis CEe	2017
PS68	Autonomic Identity Framework for the Internet of Things	Autonomic identity	Zhu, Xiaoyang and Badr, Youakim and Pacheco, Jesus and Hariri, Salim	2017
PS69	Blockchain based credibility verification method for IoT entities	Credibilty verification verification	Qu, Chao and Tao, Ming and Zhang, Jie and Hong, Xiaoyu and Yuan, Ruifen	2018
PS70	Blockchain based data integrity service framework for IoT data	Data integrity services	Liu, Bin and Yu, Xiao Liang and Chen, Shiping and Xu, Xiwei and Zhu, Liming	2017
PS71	Blockchain Connected Gateway for BLE-Based Devices in the Internet of Things	Gateway for BLE	Cha, Shi-Cho and Chen, Jyun-Fu and Su, Chunhua and Yeh, Kuo-Hui	2018
PS72	Blockchain-based dynamic key management for heterogeneous intelligent transportation systems	Key management for transportation	Lei, Ao and Cruickshank, Haitham and Cao, Yue and Asuquo, Philip and Ogah, Chibueze P Anyigor and Sun, Zhili	2018
PS73	Blockchain-based fair three-party contract signing protocol for fog computing	Three-party contract for fog	Huang, Hui and Li, Kuan-Ching and Chen, Xiaofeng	2018
PS74	FairAccess: a new Blockchain-based access control framework for the Internet of Things	FairAccess	Ouaddah, Aafaf and Abou Elkalam, Anas and Ait Ouahman, Abdellah	2016
PS75	Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT	Blockchain Meets IoT	Novo, Oscar	2018
PS76	Toward a blockchain cloud manufacturing system as a peer to peer distributed network platform	Cloud manufacturing	Li, Zhi and Barenji, Ali Vatankhah and Huang, George Q	2018
PS77	Internet of Smart Things - IoST: Using Blockchain and CLIPS to Make Things Autonomous	IoST	Samaniego, Mayra and Deters, Ralph	2017
PS78	Using blockchain to push software-defined IoT components onto edge hosts	Software-defined IoT components	Samaniego, Mayra and Deters, Ralph	2016
PS79	Bubbles of Trust: A decentralized blockchain-based authentication system for IoT	Bubbles of Trust	Hammi, Mohamed Tahar and Hammi, Badis and Bellot, Patrick and Serhrouchni, Ahmed	2018
PS80	IoT data privacy via blockchains and IPFS	IoT data privacy	Ali, Muhammad Salek and Dolui, Koustabh and Antonelli, Fabio	2017

(Continued)

Table 8. Continued

ID	Title	Short name	Author(s)	Year
PS81	The IoT electric business model: Using blockchain technology for the internet of things	IoT electric business model	Zhang, Yu and Wen, Jiangtao	2017
PS82	Using Ethereum Blockchain in Internet of Things: A Solution for Electric Vehicle Battery Refueling	Blockchain for Electrical Vehicles	Sun, Haoli and Hua, Song and Zhou, Ence and Pi, Bingfeng and Sun, Jun and Yamashita, Kazuhiro	2018
PS83	Using Blockchain for IOT Access Control and Authentication Management	IOT Access Control and Authentication	Ourad, Abdallah Zoubir and Belgacem, Boutheyna and Salah, Khaled	2018
PS84	Decentralized, blockchain based access control framework for the heterogeneous internet of things	Blockchain based access control	Dukkipati, Chethana and Zhang, Yunpeng and Cheng, Liang Chieh	2018
PS85	Mind my value: A decentralized infrastructure for fair and trusted IoT data trading	Mind my value	Missier, Paolo and Bajoudah, Shaimaa and Capossele, Angelo and Gaglione, Andrea and Nati, Michele	2017
PS86	Toward open manufacturing	Toward open manufacturing	Li, Zhi and Wang, WM and Liu, Guo and Liu, Layne and He, Jiadong and Huang, GQ	2018
PS87	Toward a robust security paradigm for bluetooth low energy-based smart objects in the Internet-of-Things	Security paradigm for bluetooth	Cha, Shi-Cho and Yeh, Kuo-Hui and Chen, Jyun-Fu	2017
PS88	Smart contract-based access control for the internet of things	Smart contract-based access control	Zhang, Yuanyu and Kasahara, Shoji and Shen, Yulong and Jiang, Xiaohong and Wan, Jianxiong	2018
PS89	Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles	Creditcoin	Li, Lun and Liu, Jiqiang and Cheng, Lichen and Qiu, Shuo and Wang, Wei and Zhang, Xiangliang and Zhang, Zonghua	2018
PS90	Patch transporter: Incentivized, decentralized software patch system for WSN and IoT environments	Patch transporter	Lee, JongHyup	2018
PS91	A sustainable home energy prosumer-chain methodology with energy tags over the blockchain	Home energy prosumer-chain	Park, Lee and Lee, Sanghoon and Chang, Hangbae	2018
PS92	A hardware-based caching system on FPGA NIC for Blockchain	A hardware-based caching	Sakakibara, Yuma and Morishima, Shin and Nakamura, Kohei and Matsutani, Hiroki	2018
PS93	Semantic blockchain to improve scalability in the internet of things	Semantic blockchain	Ruta, Michele and Scioscia, Floriano and Ieva, Saverio and Capurso, Giovanna and Di Sciascio, Eugenio	2017
PS94	Beekeeper: A blockchain-based iot system with secure storage and homomorphic computation	Beekeeper	Zhou, Lijing and Wang, Licheng and Sun, Yiru and Lv, Pin	2018

(Continued)

Table 8. Continued

ID	Title	Short name	Author(s)	Year
PS95	Blockchain based decentralized management of demand response programs in smart energy grids	Decentralized management of demand response	Pop, Claudia and Cioara, Tudor and Antal, Marcel and Anghel, Ionut and Salomie, Ioan and Bertoncini, Massimo	2018
PS96	Smart-toy-edge-computing-oriented data exchange based on blockchain	Smart-toy-edge-computing	Yang, Jian and Lu, Zhihui and Wu, Jie	2018
PS97	A blockchain-based Trust System for the Internet of Things	A blockchain-based Trust	Di Pietro, Roberto and Salleras, Xavier and Signorini, Matteo and Waisbard, Erez	2018
PS98	Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities	Privacy-preserving	Guan, Zhitao and Si, Guanlin and Zhang, Xiaosong and Wu, Longfei and Guizani, Nadra and Du, Xiaojiang and Ma, Yinglong	2018
PS99	Continuous patient monitoring with a patient centric agent: A block architecture	Continuous patient monitoring	Uddin, Md Ashraf and Stranieri, Andrew and Gondal, Iqbal and Balasubramanian, Venki	2018
PS100	Blockchain-oriented coalition formation by cps resources: Ontological approach and case study	Blockchain-oriented coalition	Kashevnik, Alexey and Teslya, Nikolay	2018

REFERENCES

[1] M. Alharby, A. Aldweesh, and A. van Moorsel. 2018. Blockchain-based smart contracts: A systematic mapping study of academic research. In *2018 International Conference on Cloud Computing, Big Data and Blockchain (ICCBB'18)*. IEEE, 1–6.

[2] Muhammad Salek Ali, Massimo Vecchio, Miguel Pincheira, Koustabh Dolui, Fabio Antonelli, and Mubashir Husain Rehmani. 2018. Applications of blockchains in the internet of things: A comprehensive survey. *IEEE Commun. Surv. Tutor.* 21, 2 (2018), 1676–1717.

[3] Luigi Atzori, Antonio Iera, and Giacomo Morabito. 2010. The internet of things: A survey. *Comput. Netw.* 54, 15 (2010), 2787–2805.

[4] Marcella Atzori. 2017. Blockchain-based architectures for the internet of things: A survey. Available at SSRN 2846810.

[5] Arshdeep Bahga and Vijay K Madisetti. 2016. Blockchain platform for industrial internet of things. *J. Softw. Eng. Appl.* 9, 10 (2016), 533.

[6] Ahmed Banafa. 2016. IoT standardization and implementation challenges. *IEEE IoT Newslett.* (2016), 1–10.

[7] Mario Barbacci, Mark H. Klein, Thomas A. Longstaff, and Charles B. Weinstock. 1995. *Quality Attributes.* Technical Report. PIttsburge Software Engineering Institute, Carnegie-Mellon University, Pittsburge, PA.

[8] Len Bass, Paul Clements, and Rick Kazman. 2012. *Software Architecture in Practice* (3rd ed.). Addison-Wesley Professional.

[9] Farokh Bastani, Wei Zhu, Hessam Moeini, San-Yih Hwang, Yuqun Zhang, et al. 2018. Service-oriented IoT modeling and its deviation from software services. In *Proceedings of the 2018 IEEE Symposium on Service-Oriented System Engineering (SOSE'18)*. IEEE, 40–47.

[10] Alessio Botta, Walter De Donato, Valerio Persico, and Antonio Pescapé. 2016. Integration of cloud computing and internet of things: A survey. *Fut. Gener. Comput. Syst.* 56 (2016), 684–700.

[11] Humberto Cervantes and Rick Kazman. 2016. *Designing Software Architectures: A Practical Approach.* Addison-Wesley.

[12] Konstantinos Christidis and Michael Devetsikiotis. 2016. Blockchains and smart contracts for the internet of things. *IEEE Access* 4 (2016), 2292–2303.

[13] Marco Conoscenti, Antonio Vetro, and Juan Carlos De Martin. 2016. Blockchain for the internet of things: A systematic literature review. In *Proceedings of the 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA'16)*. IEEE, 1–6.

[14] Hong-Ning Dai, Zibin Zheng, and Yan Zhang. 2019. Blockchain for internet of things: A survey. *IEEE IoT J.* 6, 5 (2019), 8076–8094. DOI: https://doi.org/10.1109/JIOT.2019.2920987

[15] Zheng Dong, Cong Liu, Soroush Bateni, Zelun Kong, Liang He, Lingming Zhang, Ravi Prakash, and Yuqun Zhang. 2019. A general analysis framework for soft real-time tasks. *IEEE Trans. Parallel Distrib. Syst.* 30, 6 (2019), 1222–1237. DOI: https://doi.org/10.1109/TPDS.2018.2884980

[16] Ali Dorri, Salil S. Kanhere, and Raja Jurdak. 2017. Towards an optimized blockchain for IoT. In *Proceedings of the 2nd International Conference on Internet-of-Things Design and Implementation.* ACM, 173–178.

[17] Ali Dorri, Salil S. Kanhere, Raja Jurdak, and Praveen Gauravaram. 2017. 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'17).* IEEE, 618–623.

[18] Jacob Eberhardt and Stefan Tai. 2017. On or off the blockchain? Insights on off-chaining computation and data. In *Proceedings of the European Conference on Service-Oriented and Cloud Computing*. Springer, 3–15.

[19] Dave Evans. 2011. The internet of things: How the next evolution of the internet is changing everything. *CISCO White Paper* 1, 2011 (2011), 1–11.

[20] Tiago M. Fernández-Caramés and Paula Fraga-Lamas. 2018. A review on the use of blockchain for the internet of things. *IEEE Access* 6 (2018), 32979–33001.

[21] Mohamed Amine Ferrag, Makhlouf Derdour, Mithun Mukherjee, Abdelouahid Derhab, Leandros Maglaras, and Helge Janicke. 2018. Blockchain technologies for the internet of things: Research issues and challenges. *IEEE IoT J.* 6, 2 (2018), 2188–2204.

[22] J. Gartner. 2013. Gartner predictions for IoT. Retreived on 19 July, 2018 from http://www.gartner.com/newsroom/id/2636073.

[23] Jeff Garzik. 2015. Public versus private blockchains. BitFury Group, White Paper 1 (2015).

[24] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. 2013. Internet of things (IoT): A vision, architectural elements, and future directions. *Fut. Gener. Comput. Syst.* 29, 7 (2013), 1645–1660.

[25] Mohamed Tahar Hammi, Badis Hammi, Patrick Bellot, and Ahmed Serhrouchni. 2018. Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. *Comput. Secur.* 78 (2018), 126–142.

[26] Sarra Hammoudi, Zibouda Aliouat, and Saad Harous. 2018. Challenges and research directions for internet of things. *Telecommun. Syst.* 67, 2 (2018), 367–385.

[27] Neil B. Harrison and Paris Avgeriou. 2010. How do architecture patterns and tactics interact? A model and annotation. *J. Syst. Softw.* 83, 10 (2010), 1735–1758.

[28] Anne-Wil Harzing. 2010. *The Publish or Perish Book.* Tarma Software Research Pty Limited.

[29] Jordi Herrera-Joancomartí and Cristina Pérez-Solà. 2016. Privacy in bitcoin transactions: New challenges from blockchain scalability solutions. In *Proceedings of the International Conference on Modeling Decisions for Artificial Intelligence.* Springer, 26–44.

[30] N. Kshetri. 2017. Can blockchain strengthen the Internet of Things? *IT Professional* 19, 4 (2017), 68–72.

[31] Zach Kirsch and Ming Chow. 2015. Quantum Computing: The Risk to Existing Encryption Methods. Retrieved from http://www. cs. tufts. edu/comp/116/archive/fall2015/zkir sch.pdf.

[32] Barbara Kitchenham and Pearl Brereton. 2013. A systematic review of systematic review process research in software engineering. *Inf. Softw. Technol.* 55, 12 (2013), 2049–2075.

[33] Barbara Kitchenham and Stuart Charters. 2007. Guidelines for performing systematic literature reviews in software engineering. 5 (2007).

[34] Boohyung Lee and Jong-Hyouk Lee. 2017. Blockchain-based secure firmware update for embedded devices in an internet of things environment. *J. Supercomput.* 73, 3 (2017), 1152–1167.

[35] Grace Lewis and Patricia Lago. 2015. Architectural tactics for cyber-foraging: Results of a systematic literature review. *J. Syst. Softw.* 107 (2015), 158–186.

[36] Zexin Li, Yuqun Zhang, Ao Ding, Husheng Zhou, and Cong Liu. 2020. Efficient algorithms for task mapping on heterogeneous CPU/GPU platforms for fast completion time. *J. Syst. Arch.* 114 (2020), 101936. DOI: https://doi.org/10.1016/j.sysarc.2020.101936

[37] Chun-Feng Liao, Sheng-Wen Bao, Ching-Ju Cheng, and Kung Chen. 2017. 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'17).* IEEE, 351–352.

[38] Chun-Feng Liao, Chien-Che Hung, and Kung Chen. 2019. Blockchain and the internet of things: A software architecture perspective. In *Business Transformation through Blockchain.* Springer, 53–75.

[39] Sin Kuang Lo, Yue Liu, Su Yen Chia, Xiwei Xu, Qinghua Lu, Liming Zhu, and Huansheng Ning. 2019. Analysis of blockchain solutions for IoT: A systematic literature review. *IEEE Access* 7 (2019), 58822–58835. DOI: https://doi.org/10.1109/ACCESS.2019.2914675

[40] Qinghua Lu and Xiwei Xu. 2017. Adaptable blockchain-based systems: A case study for product traceability. *IEEE Softw.* 34, 6 (2017), 21–27.

[41] Loi Luu, Viswesh Narayanan, Chaodong Zheng, Kunal Baweja, Seth Gilbert, and Prateek Saxena. 2016. A secure sharding protocol for open blockchains. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 17–30.

[42] Imran Makhdoom, Mehran Abolhasan, and Wei Ni. 2018. Blockchain for IoT: The challenges and away forward. In *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications, Volume 2 (SECRYPT'18)*. INSTICC.

[43] Henry Muccini and Mahyar Tourchi Moghaddam. 2018. Iot architectural styles. In *Proceedings of the European Conference on Software Architecture*. Springer, 68–85.

[44] Jianbing Ni, Kuan Zhang, Xiaodong Lin, and Xuemin Sherman Shen. 2017. Securing fog computing for internet of things applications: Challenges and solutions. *IEEE Commun. Surv. Tutor.* 20, 1 (2017), 601–628.

[45] Aafaf Ouaddah, Anas Abou Elkalam, and Abdellah Ait Ouahman. 2016. FairAccess: A new blockchain-based access control framework for the internet of things. *Secur. Commun. Netw.* 9, 18 (2016), 5943–5964.

[46] Alfonso Panarello, Nachiket Tapas, Giovanni Merlino, Francesco Longo, and Antonio Puliafito. 2018. Blockchain and iot integration: A systematic survey. *Sensors* 18, 8 (2018), 2575.

[47] Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz. 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. *Inf. Softw. Technol.* 64 (2015), 1–18.

[48] Otto Julio Ahlert Pinno, Andre Ricardo Abed Gregio, and Luis CE De Bona. 2017. Controlchain: Blockchain as a central enabler for access control authorizations in the iot. In *Proceedings of the IEEE Global Communications Conference (GLOBECOM'17)*. IEEE, 1–6.

[49] Joseph Poon and Vitalik Buterin. 2017. Plasma: Scalable autonomous smart contracts. White Paper. 1–47.

[50] Han Qiu, Meikang Qiu, Gerard Memmi, Zhong Ming, and Meiqin Liu. 2018. A dynamic scalable blockchain based communication architecture for IoT. In *Proceedings of the International Conference on Smart Blockchain.* Springer, 159–166.

[51] Mohammad Abdur Razzaque, Marija Milojevic-Jevric, Andrei Palade, and Siobhán Clarke. 2016. Middleware for internet of things: A survey.*IEEE IoT J.* 3, 1 (2016), 70–95.

[52] Ana Reyna, Cristian Martín, Jaime Chen, Enrique Soler, and Manuel Díaz. 2018. On blockchain and its integration with IoT. challenges and opportunities. *Fut. Gener. Comput. Syst.* 88 (2018), 173–190.

[53] Gokhan Sagirlar, Barbara Carminati, Elena Ferrari, John D. Sheehan, and Emanuele Ragnoli. 2018. Hybrid-iot: Hybrid blockchain architecture for internet of things-pow sub-blockchains. In *Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings'18) and IEEE Green Computing and Communications (GreenCom'18) and IEEE Cyber, Physical and Social Computing (CPSCom'18) and IEEE Smart Data (SmartData'18).* IEEE, 1007–1016.

[54] Mayra Samaniego and Ralph Deters. 2016. Blockchain as a service for IoT. In *Proceedings of the 2016 IEEE International Conference on Internet of Things (iThings'16) and IEEE Green Computing and Communications (GreenCom'16) and IEEE Cyber, Physical and Social Computing (CPSCom'16) and IEEE Smart Data (SmartData'16)*. IEEE, 433–436.

[55] Bilal Shabandri and Piyush Maheshwari. 2019. Enhancing IoT security and privacy using distributed ledgers with IOTA and the tangle. In *Proceedings of the 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN'19)*. IEEE, 1069–1075.

[56] Hossein Shafagh, Lukas Burkhalter, Anwar Hithnawi, and Simon Duquennoy. 2017. Towards blockchain-based auditable storage and sharing of IoT data. In *Proceedings of the 2017 on Cloud Computing Security Workshop*. ACM, 45–50.

[57] Pradip Kumar Sharma, Mu-Yen Chen, and Jong Hyuk Park. 2017. A software defined fog node based distributed blockchain cloud architecture for IoT. *IEEE Access* 6 (2017), 115–124.

[58] Nidhiben Solanki, Yongtao Huang, I.-Ling Yen, Farokh B. Bastani, and Yuqun Zhang. 2018. Resource and role hierarchy based access control for resourceful systems. In *Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC'18), Volume 2*. IEEE Computer Society, 480–486. DOI: https://doi.org/10.1109/COMPSAC.2018.10280

[59] Alexandru Stanciu. 2017. Blockchain based distributed control system for edge computing. In *Proceedings of the 2017 21st International Conference on Control Systems and Computer Science (CSCS'17)*. IEEE, 667–671.

[60] Shuai Wang, Liwei Ouyang, Yong Yuan, Xiaochun Ni, Xuan Han, and Fei-Yue Wang. 2019. Blockchain-enabled smart contracts: Architecture, applications, and future trends. *IEEE Trans. Syst. Man Cybernet.: Syst.* 49, 11 (2019), 2266–2277.

[61] Florian Wessling, Christopher Ehmke, Ole Meyer, and Volker Gruhn. 2019. Towards blockchain tactics: Building hybrid decentralized software architectures. In *Proceedings of the 2019 IEEE International Conference on Software Architecture Companion (ICSA-C'19)*. IEEE, 234–237.

[62] Claes Wohlin. 2014. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*. Citeseer, 38.

[63] Longfei Wu, Xiaojiang Du, Wei Wang, and Bin Lin. 2018. An out-of-band authentication scheme for internet of things using blockchain technology. In *Proceedings of the 2018 International Conference on Computing, Networking and Communications (ICNC'18)*. IEEE, 769–773.

[64] Mingyuan Wu, Yicheng Ouyang, Husheng Zhou, Lingming Zhang, Cong Liu, and Yuqun Zhang. 2020. Simulee: Detecting CUDA synchronization bugs via memory-access modeling. In *Proceedings of the 42nd International Conference on Software Engineering (ICSE'20)*, Gregg Rothermel and Doo-Hwan Bae (Eds.). ACM, 937–948. DOI: https://doi.org/10.1145/3377811.3380358

[65] Mingli Wu, Kun Wang, Xiaoqin Cai, Song Guo, Minyi Guo, and Chunming Rong. 2019. A comprehensive survey of blockchain: from theory to IoT applications and beyond. *IEEE IoT J.* 6, 5 (2019), 8114–8154.

[66] Mingli Wu, Kun Wang, Xiaoqin Cai, Song Guo, Minyi Guo, and Chunming Rong. 2019. A comprehensive survey of blockchain: From theory to IoT applications and beyond. *IEEE IoT J.* 6, 5 (2019), 8114–8154. DOI: https://doi.org/10.1109/JIOT.2019.2922538

[67] Mingyuan Wu, Lingming Zhang, Cong Liu, Shin Hwei Tan, and Yuqun Zhang. 2019. Automating CUDA synchronization via program transformation. In *Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering (ASE'19)*. IEEE, 748–759. DOI: https://doi.org/10.1109/ASE.2019.00075

[68] Q. I. Xia, Emmanuel Boateng Sifah, Kwame Omono Asamoah, Jianbin Gao, Xiaojiang Du, and Mohsen Guizani. 2017. MeDShare: Trust-less medical data sharing among cloud service providers via blockchain. *IEEE Access* 5 (2017), 14757–14767.

[69] Fangmin Xu, Fan Yang, Chenglin Zhao, and Chao Fang. 2018. Edge computing and caching based blockchain IoT network. In *Proceedings of the 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN'18)*. IEEE, 238–239.

[70] Xiwei Xu, Cesare Pautasso, Liming Zhu, Vincent Gramoli, Alexander Ponomarev, An Binh Tran, and Shiping Chen. 2016. The blockchain as a software connector. In *Proceedings of the 2016 13th Working IEEE/IFIP Conference on Software Architecture (WICSA'16)*. IEEE, 182–191.

[71] Xiwei Xu, Cesare Pautasso, Liming Zhu, Qinghua Lu, and Ingo Weber. 2018. A pattern collection for blockchain-based applications. In *Proceedings of the 23rd European Conference on Pattern Languages of Programs*. ACM, 3.

[72] Xiwei Xu, Ingo Weber, and Mark Staples. 2019. *Architecture for Blockchain Applications*. Springer.

[73] Xiwei Xu, Ingo Weber, Mark Staples, Liming Zhu, Jan Bosch, Len Bass, Cesare Pautasso, and Paul Rimba. 2017. A taxonomy of blockchain-based systems for architecture design. In *Proceedings of the 2017 IEEE International Conference on Software Architecture (ICSA'17)*. IEEE, 243–252.

[74] Wendy Yánez, Redowan Mahmud, Rami Bahsoon, Yuqun Zhang, and Rajkumar Buyya. 2020. Data allocation mechanism for internet-of-things systems with blockchain. *IEEE IoT J.* 7, 4 (2020), 3509–3522. DOI: https://doi.org/10.1109/JIOT.2020.2972776

[75] Wendy Yánez, Redowan Mahmud, Rami Bahsoon, Yuqun Zhang, and Rajkumar Buyya. 2020. Data allocation mechanism for internet-of-things systems with blockchain. *IEEE IoT J.* 7, 4 (2020), 3509–3522.

[76] I.-Ling Yen, Farokh B. Bastani, Yongtao Huang, Yuqun Zhang, and Xin Yao. 2017. SaaS for automated job performance appraisals using service technologies and big data analytics. In *Proceedings of the 2017 IEEE International Conference on Web Services (ICWS'17)*, Ilkay Altintas and Shiping Chen (Eds.). IEEE, 412–419. DOI: https://doi.org/10.1109/ICWS.2017.47

[77] I.-Ling Yen, Shuai Zhang, Farokh Bastani, and Yuqun Zhang. 2017. A framework for IoT-based monitoring and diagnosis of manufacturing systems. In *Proceedings of the 2017 IEEE Symposium on Service-Oriented System Engineering (SOSE'17)*. IEEE, 1–8.

[78] Q. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani. 2017. Medshare: Trustless medical data sharing among cloud service providers via blockchain. *IEEE Access* 5 (2017), 14 757–14 767.

[79] Dongjin Yu, Yike Jin, Yuqun Zhang, and Xi Zheng. 2019. A survey on security issues in services communication of microservices-enabled fog applications. *Concurr. Comput. Pract. Exp.* 31, 22 (2019). DOI: https://doi.org/10.1002/cpe.4436

[80] Mengshi Zhang, Yuqun Zhang, Lingming Zhang, Cong Liu, and Sarfraz Khurshid. 2018. DeepRoad: GAN-based metamorphic testing and input validation framework for autonomous driving systems. In *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering (ASE'18)*, Marianne Huchard, Christian Kästner, and Gordon Fraser (Eds.). ACM, 132–142. DOI: https://doi.org/10.1145/3238147.3238187

[81] Tianlei Zheng, Yuqun Zhang, Xi Zheng, Min Fu, and Xiao Liu. 2017. BigVM: A multi-layer-microservice-based platform for deploying saas. In *Proceedings of the 5th International Conference on Advanced Cloud and Big Data (CBD'17)*. IEEE Computer Society, 45–50. DOI: https://doi.org/10.1109/CBD.2017.16

[82] Tianlei Zheng, Xi Zheng, Yuqun Zhang, Yao Deng, ErXi Dong, Rui Zhang, and Xiao Liu. 2019. SmartVM: a SLA-aware microservice deployment framework. *World Wide Web* 22, 1 (2019), 275–293. DOI: https://doi.org/10.1007/s11280-018-0562-5

[83] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. 2017. An overview of blockchain technology: Architecture, consensus, and future trends. In *Proceedings of the 2017 IEEE International Congress on Big Data (BigData Congress'17)*. IEEE, 557–564.

[84] Husheng Zhou, Wei Li, Zelun Kong, Junfeng Guo, Yuqun Zhang, Bei Yu, Lingming Zhang, and Cong Liu. 2020. DeepBillboard: Systematic physical-world testing of autonomous driving systems. In *Proceedings of the 42nd International Conference on Software Engineering (ICSE'20)*, Gregg Rothermel and Doo-Hwan Bae (Eds.). ACM, 347–358. DOI: https://doi.org/10.1145/3377811.3380422

[85] Michael Herbert Ziegler, Marcel Groβmann, and Udo R. Krieger. 2019. Integration of fog computing and blockchain technology using the plasma framework. In *Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC'19)*. IEEE, 120–123.

Received August 2020; revised October 2020; accepted December 2020