Blockchain-based trusted data sharing among trusted stakeholders in IoT

Peichang Shi1 | Huaimin Wang1 | Shangzhi Yang1 | Chang Chen2 | Wentao Yang2,3

1College of Computer Science, National University of Defense Technology, Changsha, China
2Ziggurat Technology, Xi’an, China
3School of Economics and Management, Tsinghua University, Beijing 100084, China

Correspondence
Peichang Shi, College of Computer Science, National University of Defense Technology, Changsha 410073, China. Email: pcshi.nudt@gmail.com

Funding information
National Key R&D Program of China, Grant/Award Number: 2016YFB1000100; National Natural Science Foundation of China, Grant/Award Number: 61772030; GF Innovative Research Program

Summary
Sharing trusted data among trusted stakeholders is very important to large-scale Internet of Things (IoT) applications. However, the entities and organizations involved in IoT naturally lack trusted relationships, which poses significant challenges to the above vision. Specifically, the first challenge is to ensure that the data in the physical world can be objectively and truly injected into the information world of IoT. The second is to ensure the credibility of the entities’ identities in IoT. The third is to ensure the authenticity of data, the credibility of identity, and the reliable transmission of data when a third trusted party is unable to provide the expected trusted services. In view of the above challenges, this paper proposes a secure and lightweight triple-trusting architecture (SLTA), which fully uses a blockchain-related supporting technology. The architecture includes an oracle-based data collection mechanism, which ensures that the data collected from edge devices of IoT cannot be modified, and the distributed identity management mechanism, which enhances personal privacy, security, and control of digital identities. Furthermore, a series of innovative designs for applying the blockchain to special large-scale cooperation scenario in IoT are proposed, which is also a part of the key mechanisms of the SLTA. The innovative design includes a new software-defined blockchain structure model and a lightweight Byzantine fault-tolerant algorithm that provides credible support for decentralized data collection, identity management, and data transfer, as well as low-overhead sequential storage mechanism.

KEYWORDS
blockchain, Internet of Things, large-scale cooperation, trusted data sharing

1 | INTRODUCTION

With the development of supporting technologies and the innovation of application modes, Internet of Things (IoT) has the potential to be one of the most popular paradigms in the era of Internet computing. In IoT ecosystem, every entity equipped with identifying, sensing, networking, and processing capabilities has the ability to communicate with all other entities to accomplish some objective through a multitype communication style. Although very large-scale IoT deployments remain a few years away, IoT is large scale by nature. Medium-scale or even large-scale cooperation among smart
entities of IoT, such as smart cities, connected cars in a sharing economy, the Internet of Multimedia Things (IoMT), wireless multimedia sensor networks (WMSNs), etc, is increasingly being considered. The IoT systems cannot successfully realize the notion of pervasive connectivity and cooperation if they are incapable of accommodating the data tsunami generated by such systems and sharing data in an efficient and trusted way. In other words, IoT systems should have the ability for trusted sharing of trusted data among trusted stakeholders.

However, the entities and organizations involved in IoT systems naturally lack trusted relationships, which poses three challenges to the above vision:

- How to ensure that the authentic data collected by physical sensors can be transmitted without being tampered with before injected into the information world of IoT.
- How to ensure the credibility of an entity’s identity, i.e., the entity’s identity in IoT is real and trusted.
- How to ensure the authenticity of data, the credibility of identity, and the reliable transmission of data when a trusted third party is unable to provide the expected trusted services.

In reality, the operation of IoT systems usually involves a trusted third party as the central entity that executes decisions and performs scheduling. The existence of a trusted third party results in the four following challenges. First, the cumbersome process may affect efficiency. Second, there is a risk of the central entity’s failure, which may lead to unreliability of the system. Third, the existence of trusted third parties often means that IoT systems adopt a central hierarchy, with the two objectives that the higher level of the hierarchy be more comprehensive and that the lower level be more detail oriented, which are unattainable at the same time. Fourth, the cost of maintaining and changing the central hierarchy is very high.

In view of the above dilemma, this paper proposes a secure and lightweight triple-trusting architecture (SLTA), which fully uses a blockchain-related supporting technology. Blockchain’s key characteristics, such as decentralization, persistence, anonymity, and auditability, can facilitate establishing trusted relationships among untrusted entities in a decentralized environment. In our previous work, JointCloud was proposed as a new cross-cloud cooperation paradigm of the future cloud computing, using blockchain to ensure the credibility and auditability of collaboration services. In the large-scale IoT collaboration scenario, it is novel to propose a blockchain-based solution to such problems as data confirmation, trusted endorsement of sources, consensus on important information, and trusted management of individual identity. However, in a large-scale IoT collaboration scenario, there are such characteristics as a dynamic change in a cooperative relationship, a change in participating nodes, and elasticity of scale, as well as diversity and heterogeneity of nodes in computing, storage, communication, and other aspects. This also makes applying blockchain technology to this scenario a significant technical challenge. Hence, the contributions of this paper are mainly reflected in the following two aspects:

1. We propose an SLTA. The SLTA includes an oracle-based data collection mechanism, which ensures that the data collected from edge devices of IoT cannot be modified, and the distributed identity (DID) management mechanism, which enhances the personal privacy, security, and control of digital identities.
2. We propose a series of innovative designs for applying blockchain in the special large-scale IoT cooperation scenario, which is also a part of the key mechanisms of the SLTA. The technological innovations include a new software-defined blockchain structure model, a lightweight Byzantine fault-tolerant algorithm that provides credible support for decentralized data collection, identity management, and data transfer, a low-overhead sequential storage mechanism, etc.

The rest of this paper is organized as follows. Section 2 proposes an SLTA. Section 3 presents the key mechanisms of the SLTA. Section 4 reviews the related work on IoT and blockchain in IoT. Finally, conclusions are provided in Section 5.
passive partition, structure disturbance caused by the disconnection of fixed nodes and mobile nodes, arbitrary changes in the number of nodes, and nodes heterogeneity in computing, storage, network, and other capacities. These are specialized and sophisticated questions; as a result, application of the traditional blockchain technology to new scenarios requires new customizations. The two most commonly used blockchains on the Internet, the consortium blockchain and the public blockchain, have their own advantages and disadvantages. For example, consortium blockchain cannot very well support dynamic joining and exits of nodes, whereas the public blockchain lacks a trusted access control mechanism. In the field of IoT, there are open questions of how to make a large number of nodes form a chain by themselves, how to ensure access control, and how to exchange data across chains. This paper’s goal is to learn from the technical advantages of the flexible networking ability of the public blockchain, the nodes and access control and authority management of the consortium blockchain, and to present the SLTA that supports trusted data collection and validation, trusted identity management, and trusted data transmission and sharing.

The SLTA is based on a triangle of consensus, authorization, and chain storage. The four fundamental supporting mechanisms surrounding these three main points of trusted data sharing mainly involve physical real data injection, DID management and access authorization, a low cost sequential storage model, and a lightweight and efficient Byzantine fault tolerant (BFT) consensus algorithm. As shown in Figure 1, the data of the physical world are endorsed by one of the three types of oracle (software, hardware, or consensus), injected into the user contract as trusted data and cached in the nodes’ storage system with the low cost sequential storage model. The lightweight and efficient BFT consensus algorithm is called to confirm the ownership of the data, perform the identity authorization, and build a trust relationship that does not rely on a central hierarchy. The related information is written into the blockchain at the same time. The DID management and authorized access mechanism help manage and state identifications and realize data access control.

In this architecture, data injection is supervised by an oracle. A blockchain is a closed environment in the information world, which cannot ensure that the data obtained from the physical world are objective and trustworthy. To address this limitation, the oracle is used to validate the data when the data from the physical world enter the information world. There are three kinds of oracles: software, hardware, and consensus. The first two are centralized and the last is achieved by distributed consensus mechanism. The main function of the oracle machine is smart contract and external data interaction,
which supports the real data input from the real world. In other words, it is the channel that links the blockchain world to the real world. An oracle needs to deploy an oracle’s smart contract. If the data access service is required, the user must include the oracle’s contract in the user’s own smart contract and call such services via related APIs. However, how does the oracle prove that the data it receives from the physical world are objective and trustworthy? The main proof of trust mechanism it relies on is transport layer security (TLS) certification technology on the basis of the protocol TLS 1.1, which is used to provide confidentiality and data integrity between two communication applications. The biggest advantage of the mechanism is its independence from the application protocol, and higher-level protocols can be used transparently on top of the TLS protocol. The TLS proof mechanism includes three parts: the server, the auditee, and the auditor. The server part provides data from the physical world, the auditee part is the oracle contract, and the auditor exists as an open source instance that mainly reviews and verifies the data provided by the oracle in the past to ensure the integrity and security of data. The IoT nodes register and save their identity (ID) and statement in blockchain, which enables any two users from various application scenarios to verify each other without a trusted third party. Limited by IoT nodes’ storage capacity and energy constraints, SLTA provides a scientific storage strategy. Each node will store blocks with high value density or recently generated blocks. Blocks with low value density or earlier generated blocks will be stored in remote or back-end cloud through historical block management. Blocks with medium value density can be divided into many slices according to a low cost storage model and each node stores only a part of them, which helps reduce the total storage. As to the consensus algorithms in the SLTA, a scientific trade-off must be made among three characteristics of security, efficiency, and fairness (discussed in Section 3.3.2). Different scenarios require the selection of consensus algorithms, eg, the practical Byzantine fault tolerance (PBFT),9 Zyzzyva,10 and Proof of Work (PoW),11 which have different characteristics. Lightweight, high efficiency, and the ability of Byzantine fault tolerance are the basic demand standardization in the IoT scenario.

3 | KEY MECHANISMS OF SLTA

The challenges of building such an architecture are guaranteeing that the data collected by sensors are authentic and not tampered with, ensuring that the entities involved are trustworthy, and most importantly, remaining stable when third parties are unable to provide the expected services. By applying two trusted mechanisms and implementing three technology innovations, SLTA is able to conquer above challenges. The first two innovations leverage the oracle and DID technologies to ensure data authenticity, data credibility, and access authorization. The latter mainly provides three blockchains for the technological innovation of IoT and provides customized optimization for the blockchain application in the large-scale collaboration scenario.

3.1 | Oracle-based data collection mechanism

A cryptographic method is the basis for ensuring the reliable collection and secure data injection into the blockchain. Each account of the blockchain is the exact key that ensures data security and integrity by cryptographic methods; the design also ensures that the blockchain can be a self-organizing, self-driven, and decentralized system. For any account-based data system, the most important issue is security and credibility.

Security is the most basic and vital requirement in an account system. First, because the blockchain ledger has certain transparency, all consensus nodes are required to confirm each transaction and reach a consensus. In this case, the traditional username-password system is unsuitable for the distributed application on the blockchain. Hence, a decentralized account system that relies on cryptographic algorithms such as asymmetric encryption has emerged, so that blockchain applications can ensure the data ownership of each node in an open environment. Second, since the blocks on the blockchain allow access to anyone, malicious operations are difficult to control and roll back. As a result, it is necessary to guarantee that no malicious operation can be successfully completed. The basic requirement is that the data need to be collected by authorized collectors, and the data sources need to be verifiable by communication nodes as well as any other third party. The evidence used for verification should be difficult to imitate or modify by malicious attackers. Ensuring the security of the account’s digital assets and resisting any possible attacks is the most basic requirements of the blockchain data account system.

Since the account system is closely related to data flow and sharing, the decentralized account system should also support highly concurrent data processing. The high concurrency in the decentralized account system has bottlenecks in two places. First, although the process of reaching consensus requires time, every confirmation must reach distributed consensus. Second, the account status must be modified after each block has been confirmed. If a certain data processing
is based on the current block, but the account status is computed and broadcasted to the blockchain network after the next block has arrived, it is very likely that a change in the account status has occurred, as a result, a conflict might occur. The decentralized account system should be able to handle this issue correctly and effectively, supporting high concurrency and thus enabling wider applications. Another problem with the data account system is how to secure, preserve, and apply private keys. In this architecture, using secure hardware to implement the storage and operations of the private key is planned. The hardware requirements support most of the mainstream encryption and decryption algorithms.

After the data have been collected, the next step is to guarantee that the data are transmitted and injected into the blockchain securely. Therefore, it must resist attacks and provide a mechanism for self-certification of innocence for a third party to use for verification while data are being transmitted. The oracle shown in Figure 2 is a trusted entity that introduces information of the state of the external world by signing messages and allowing the specified smart contract to react to an uncertain external world. It has the characteristics such as making data tamper-resistant and providing a stable service and auditable data, along with an economically motivated mechanism for ensuring the health of every operation.

The oracle has two realization modes: single and multiple. The multiple mode is sometimes also called the “oracle network.” A single mode contains only one oracle and that oracle is trusted and executes the code correctly, and the contract participants can be sure that it will not collude with any participant in the contract. A single mode is similar to a SaaS provider. For most applications, a single mode is already safe enough and cost effective. The multiple mode is complex and has a high cost. Typically, it is applied in fields that require higher data reliability and involve relatively large values. When data are input, the oracle network needs to ensure that data of each participant are unknowable to other participants. Then, each node inputs its own data into the smart contract. The smart contract will select the data closest to the median in the case of continuous data such as price; in the case of binary data, the votes will be counted. Finally, the network will reward the nodes that provided the correct data. Unlike the single mode, an oracle network needs to consider the Sybil attack and the collusion attack.

As an input channel for real-world information, the oracle provides a trusted external data access service for the blockchain. Through it, the external information can trigger the action in the blockchain, breaking the information barrier between the blockchain and the real world. The oracle service can help the blockchain application collect data from IoT devices and guarantee safety using the security hardware. By introducing a verification authority, without prejudice to normal network communication, it is confirmed that the service is constrained and can only send the data provided by a trusted data source. With supporting of cryptographic methods, the constraint process can be verified. Additionally,
each time the data are provided, a certification document is generated, which can be verified by any third party wishing to confirm the validity of the results and the delivery process.

### 3.2 Distributed identity management mechanism

A DID management mechanism, shown in Figure 3, follows the upcoming decentralized identifiers standard\(^\text{12}\); such identifiers are of a new type that is globally unique, resolvable with high availability, and cryptographically verifiable. A DID mechanism can complete the registration, parsing, update, and revocation operations without centralized registration and authorization, and remains independent of any centralized authority. A DID is specifically resolved as a DID document that mainly contains two aspects: one is the encryption material (such as a public key and an anonymous identity recognition protocol), and the other is attributions (including service nodes and authentication information). Authentication information and encryption materials are used in DID authentication; the service nodes support a trusted interaction with the DID entities. A verifiable credential provides a standard for describing certain attributes of entities. It can represent the same information as credentials in the physical world. The DID holders, with verifiable declarations, can prove to other entities that their attributes are trusted. Simultaneously, the combination of a digital signature and the zero-knowledge proof cryptographic technology can make a statement more secure and credible, and further protect user privacy against violation. A verifiable declaration, which can be validated or signed by other entities, is signed and issued by an entity to describe its attributions. In addition, verifiable declarations support both centralized systems and decentralized systems. In a centralized system, one or more entities are specified as a trust anchor. The trust anchor, as a well-known and absolutely trusted entity, can specify another entity as a trusted entity, and the specified trusted entity can still specify other entities as a trusted entity as well, resulting in a growing trusted network centered at trust anchors. In contrast, in a decentralized trust network with no necessity to choose a trust anchor, a trusted network is generated spontaneously by peer-to-peer (P2P) verification to establish trust relationships. In addition, the more an entity is trusted by others, the higher is its credibility.

Following the DID reference design, proposed by the Credentials Community Group (W3C),\(^\text{12}\) DID is combined with the underlying layer of the blockchain, allowing any entity on the blockchain to independently create and manage its own identity. One entity can correspond to multiple DIDs to satisfy the desire for separation of identities, personas, and
applications. In this circumstance, the entity can refer to any objective and distinguishable object in the real world, such as an individual, an organization, or a certain thing.

Generally, certification organizations can issue publicly trusted statements to general entities; examples are driver's licenses issued by a vehicle management office and student ID cards issued by schools. When these offline statements are put on the network for verification and applications, there may be problems such as time delays, data modification, or disclosure of private data. Hence, putting standardized verifiable declarations on the blockchain will make it more convenient and controllable and ease authentication. Furthermore, the zero-knowledge proof technique is added to expand functions such as DID identification, which can enable anonymous issuance of verifiable declarations or protect private data while identity is verified without personal information being exposed.

In terms of transaction performance, capacity, privacy protection, and compliance supervision, both consortium blockchain and private blockchain technologies are widely used by many enterprises. However, to some extent, this violates the decentralized value and trust system of the blockchain and prevents the digital assets in the blockchain from being transferred directly between different blockchains, which leads actively or passively to the emergence of “islands of value.” The limitations of the consortium blockchain and private blockchain have led to the application of various cross-chain technologies to connect different blockchains. At present, there are well-known cross-chain technologies such as side chain, Polkadot, and interledger.

The solution of the cross-chain interaction problem includes a blockchain layer, a cross-chain service layer, and a cross-chain application layer. By integrating a cross-chain communication protocol and cross-chain software development kit in the blockchain layer, it is possible to enable cross-chain invocation, information exchange, and asset circulation between homogeneous or heterogeneous blockchains.

### 3.3 Innovative design of blockchain applicable to SLTA

The blockchain technology reduces the degree of centralization of IoT networks that previously followed a hierarchical structure and provides a data and node verification mechanism without the use of a third party. However, due to the variety of computing, storage, and transmission capabilities of IoT devices, it is impossible to directly use the blockchain technology for IoT; instead, technological innovations have to be developed that adapt to new scenarios. To this end, this paper proposes a special analysis of IoT and designs three new technologies as follows: (1) a new software-defined blockchain structure mode, (2) a lightweight and efficient consensus algorithm, and (3) a low-overhead sequential storage model.

#### 3.3.1 New software-defined blockchain structure model

Aiming at the performance problems faced by the existing blockchain systems, the academic community has proposed various improvement schemes. For example, the BitCoin-NG system divides blocks into key blocks and microblocks without changing the block size. The key blocks are used for election and are generated every ten minutes using the PoW consensus; the microblocks are used for serialization transactions and are generated between two key blocks. The throughput of the blockchain system is increased and the transaction acknowledgment delay is reduced due to an increased number of microblocks generated during the same period of time. To some extent, BitCoin-NG is an attempt to mix organizational structures that fuse two different types of chains.

Existing graph-based blockchain systems using directed acyclic graph are often difficult to apply to large-scale collaborative consensus scenarios due to the large communication overhead. Under current leading optimization algorithms such as HashGraph, participating nodes are configured with a transaction chain that records their behavior. Accordingly, the concept of HashGraph is used to manage the interaction between nodes. Each node can calculate locally the global ordering between transactions, forming a graph structure of the overall system, so that it can be performed without sending messages to each other to achieve the purpose of the confirmation. The HashGraph can be regarded as an attempt to develop a graph-based blockchain.

The traditional chain-based and graph-based blockchain system has its own advantages and disadvantages. The design of a new blockchain organization structure suitable for special scenarios and supporting the coexistence of homologous and heterogeneous subchains should combine the above two structures. The application of sharding technology in the blockchain makes such a combination possible. However, it is essential to solve the following problems: (1) traceability and compatibility between different chains with heterogeneous structure when a chain diverges to a subgraph and a
3.3.2 Lightweight and efficient consensus algorithm

The blockchain consensus algorithm is the core of maintaining a common cognition on the changing data among a large number of geographically distributed P2P nodes. Distributed system consensus algorithms such as Paxos and Raft are mostly used in scenarios where the environment is relatively reliable and the number of nodes is limited, whereas the algorithms such as PoW, POS, DPOS, PBFT, and Algorand focus more on BFT among a very large number of untrusted nodes in an open and anonymous environment. The existing blockchain consensus algorithms have their own advantages and disadvantages, yet no such algorithm can achieve optimality in all scenarios. Therefore, a comprehensive evaluation of the consensus algorithm from multiple perspectives is of great significance for the customization and optimization of various scenarios.

Based on the existing research on blockchain consensus algorithms, the design of the blockchain consensus algorithm is evaluated from three perspectives: efficiency, security, and fairness as shown in Figure 5. (1) The efficiency dimension mainly includes efficiency, scalability, and energy consumption. Efficiency is mainly reflected in the delay of a definitive or probabilistic final consensus after a block has been chained. The scalability is reflected in the speed of the block being chained and the number of transactions contained in each block, ie, the ability to process a given number of transactions per second. Energy consumption refers to the power consumption and the communication overhead in the process of reaching consensus. (2) The security dimension mainly includes forking, the Sybil attack, privacy protection, and BFT. Forks in Bitcoin lead to waste of a significant amount of computational power (algorithms such as Algorand aim to avoid...
forks). Resistance to the Sybil attack is one of the uncompromising metrics of all consensus algorithms. For privacy protection, systems such as Bitcoin pursue identity hiding and transaction exposure. However, there are many application scenarios in which there is a need for transaction not being exposed. The BFT means that the blockchain system can tolerate Byzantine fault nodes, which is also the basic difference between the blockchain consensus algorithm and the distributed consistency algorithm. (3) The fairness dimension mainly includes the degree of decentralization (eg, complete decentralization or a certain degree of centralization), free access of nodes to the blockchain network, and whether the accounting opportunities are equal. The degree of decentralization refers to the system operation phase rather than the design phase. For example, Bitcoin is strongly centralized in the design phase. Nevertheless, in the beginning of its operation, Bitcoin showed the characteristics of decentralization, whereas some analysts believe that Bitcoin has become more centralized recently. In practice, there are also numerous scenarios that do not need complete decentralization. In such scenarios, it is only necessary to increase the degree of decentralization to reduce the operating cost of the system. The consensus algorithm that supports the free entry and exit of nodes is more consistent with the actual needs of the consortium blockchain and the public blockchain. More equal competition for bookkeeping rights is a manifestation of whether the blockchain system is democratic and fair.

The existing blockchain consensus algorithm is scientifically balanced in three dimensions of effectiveness, security, and fairness to suit various scenarios. The ten metrics of three dimensions cannot be satisfied at the same time. Hence, in various scenarios, different weight of metrics are chosen, and the algorithms that satisfy more high-weight factors are adopted. Based on the intrinsic relationship model of the abovementioned metrics, guidelines need to be provided for designing the customized consensus algorithm for different demands under various scenarios. Since the existing internet consensus algorithms are difficult to apply to large-scale cooperation scenarios, based on the abovementioned association model, a new lightweight consensus algorithm will be proposed in the SLTA to further enrich the consensus algorithm set and support usage of the architecture in such scenarios.

The core problem solved by the consensus algorithm is that of the choice of nodes and the manner of reaching consensus. In this paper, we will first study a lightweight and efficient consensus algorithm with central nodes and trusted hardware. The consensus algorithm without a center will be researched in the future. The large-scale cooperation scenario involves numerous nodes, so the new consensus algorithm will form a voting committee based on trusted hardware and smart contract election. The lifetime of a voting committee depends on the evaluation of the authoritative nodes in the backend. Members of the frontend committee take turns to make the block proposal as the master node. For the high-priority data (such as control and command instructions), the master node initiates the voting process in real time, so that all members can form a common cognition of the data. At the same time, with gossip broadcast and directed propagation, the success of information reception is guaranteed; for low-priority data, the master node packs all such data during a period into blocks and initiates one voting process for all members. The size and frequency of the block is determined based on the cumulative value of the information priority.

In the process of voting, adopting a fast Byzantine fault-tolerant consensus mechanism, as shown in Figure 6, is proposed. The basic idea is that in the normal state, only $f+1$ Byzantine nodes are needed for consensus, while $f$ is defined as the maximum number of Byzantine fault nodes that a consensus algorithm can tolerate. When the primary node detects $r(r<f+1)$ Byzantine results, it dynamically enables the remaining $f$ nodes to join the consensus process. When $f+1$ nodes have reached the same result, consensus can be attained, and the primary node broadcasts the block information.
to authorized nodes via the gossip protocol. As for the f+1 nodes selection, the algorithm can consider factors such as the node's computing and communication ability. The process of view change (i.e., the change in the primary node) occurs via a periodic polling method. The re-election interval of the committee is relatively long, and the time interval depends on the evaluation result of the behavior of the frontend node by the backend authority node. The algorithm reduces the replication cost from 2f+1 to practically f+1, i.e., it is a relatively lightweight and low-overhead proposal. It can be a practical and cost-effective approach to providing the BFT ability in large-scale cooperation scenarios of IoT.

### 3.3.3 Low-overhead sequential storage model

In a large-scale IoT cooperation scenario, various local autonomous cooperative systems undertake different tasks. For a complex mission, continuous observation of the historical status inevitably leads to an increment in the cumulative data volume. Compared with nodes of the Internet, IoT nodes' storage capacity is weak and uneven. Therefore, the data accumulated by long-term tasks may exceed the storage capacity of some nodes.

In view of the above, we propose a low-overhead storage model for continuous storage of blockchain data in the large-scale cooperation scenario. The core problems of the model include the following: (1) Due to the weak and uneven storage capacity of IoT devices and the continuous growth of historic data, running out of space to store data is inevitable. (2) To perform the verification method of the blockchain, each authorized node is required to store a copy of the entire ledger. As the data size increases, the oldest data of a node with a weak storage capacity may have to be replaced by new data, which will influence the integrity of the ledger held by such a node. (3) Although in some cases the historical correlation between tasks is relatively low, it is necessary to have the ability to reproduce the historic status. Hence, from a global perspective, persistent storage of historic data is also required.

In practical applications, the acquisition time of the data can be treated as the selection criterion of the following three strategies. The first strategy is full storage, i.e., all nodes store the full current data. The second is partial storage, i.e., each node only needs to store a part of the older data, whereas complete data can be restored using the combined data of all of these nodes. The last strategy is persistent storage, i.e., the low-value density data or the oldest data need to be stored in a remote centralized database.

A low-overhead sequencing storage model solving the above dilemma is shown in Figure 7. The core mechanisms include the following. (1) A data reduction mechanism based on smart contracts: In the large-scale cooperation scenario, the data collected by the sensors in the system are first cached inside IoT equipment, and then the predefined data template is selected according to the data sources to refine key information. Therefore, the original data size can be reduced with the accompanying decrease in the storage capacity demand, which is equivalent to an increase in the storage capability of the device. (2) Delayed data slicing mechanism: To alleviate demand for data storage brought by the need to store the entire ledger, we decompose the older data of the ledger into slices according to the storage capacity of each node. If the slice method is used, the partial data are only stored on some nodes, and the storage needed for these data is reduced. In the transaction verification phase, each node reads the ledger information in the local storage, acquires the slices of other nodes through the network, and then recombines them into complete ledger data according to the original chain order, thereby achieving the verification of the transaction. This mechanism can effectively avoid the problem that the number of nodes capable of performing verification decreases with the increase in data volume. At the same time, the distributed storage of data makes it more difficult for an attacker to obtain the complete data through network intrusion, which improves the security of the autonomous system to some extent. (3) Historical data archive and persistence storage mechanism: Since the task has no historical relevance in some environments, we consider dividing the task and automatically deleting the historical storage data. However, in practice, it is often necessary to examine the entire historical
status. Therefore, we consider that, when conditions permit, the historic data could be sent to a remote data center for archiving before such data are deleted from local nodes.

4 | RELATED WORK

The IoT envisions pervasive, connected, and smart nodes interacting autonomously while offering all sorts of services.\textsuperscript{19} To enable large-scale cooperation, large-scale data collection and sharing have become the basic ability of the future wide range of IoT applications. To this end, recent research has provided significant support, especially for trusted transmission of very large volumes of data with and without a trusted third party.

The existing research has laid a foundation for large-scale cooperation of IoT nodes through transmitting data interactively. The sensors, objects, and individuals equipped with smart devices are capable of communicating with each other\textsuperscript{20} with little or even no human intervention. The WMSNs\textsuperscript{5} have been proposed, aiming to improve the data transmission capacity, and multipath routing protocols with multiconstraints have been recommended. To facilitate multimedia-based services and applications, the IoMT\textsuperscript{4} as a novel paradigm has been presented in which smart heterogeneous multimedia things can interact and cooperate with each other. In the future, the supporting architectures and technologies of WMSNs and IoMT can be used to enhance data transmission capabilities in IoT. Attribute-based encryption has been proposed as an effective cryptographic tool for secure management of IoT devices.\textsuperscript{20} The QADA\textsuperscript{21} is a hybrid Quality of Service-Aware Data Aggregation scheme, that combines the features of the cluster-based and tree-based data aggregation schemes, addresses some of their important limitations, and outperforms such schemes in terms of power consumption, network lifetime, and bearing higher traffic load. Hassanein and Oteafy\textsuperscript{22} indicate that the sheer volume and velocity of data generated by IoT systems are burdening the networking infrastructure, especially at the network edge, and advocate for a uniform view of data management in IoT systems, to gather more data for better analytics. Kumar et al\textsuperscript{23} observe that IoT should enable objects, such as smart devices, connected devices, smart buildings, etc, to have the ability to collect and exchange data. Taherkordi and Eliassen\textsuperscript{24} propose a service-oriented design architecture that is particularly focused on provisioning and processing data-centric IoT services over Fog-Cloud systems. As multimedia IoT systems are widely used, Long et al\textsuperscript{25} propose an edge computing framework to enable cooperative processing on resource-abundant mobile devices for delay-sensitive multimedia IoT tasks to support long-distance transmission of a large volume of video. These related studies represent a large amount of research and exploration of the aspects of IoT node management and data transmission security.

Research of blockchain-based security and privacy issues for large-scale IoT collaboration has made some progress. As IoT networks are widely distributed, and open and contain many nodes collecting and processing private
information, such networks are becoming a goldmine of data for malicious actors. Security issues, such as privacy, access control, secure communication, and secure storage of data, are clearly becoming significant challenges in IoT environment. However, several intrinsic features of IoT, including the lack of central control, heterogeneity of device resources, multiple attack surfaces, context-specific risks, and scale, amplify its security and privacy challenges. There has been increasing interest in adopting blockchain in IoT for security and privacy. The foundational concepts such as decentralized trust and distributed ledger are promising for distributed and large-scale IoT applications, which enable multiple stakeholders’ interactions without requiring a trusted third party. Khan and Salah present major security issues, outline security requirements of IoT, and discuss how many aspects of the challenges, such as address space, identity and access management, data authentication and integrity, authorization and privacy, and secure communications, can be solved by the blockchain. Dorri et al propose a blockchain-based smart home framework in which each smart home is equipped with an always-online device with ample resources, known as a miner, which is responsible for handling all communication within and outside the home, and also preserves a private and secure blockchain used for controlling and auditing communications, and analyzes its security with respect to the fundamental goals of confidentiality, integrity, and availability. Choi et al implement a scheme to securely control IoT devices using smart contracts, which provides guaranteed authentication, non-repudiation, and integrity without any central administration. The BIFIT is a blockchain-based identity framework that enables a smart home to achieve identity self-management by end users through autonomously extracting appliances’ signatures, creating blockchain-based identities for the appliance owners, and correlating appliances’ signatures and owners’ identities. Ayoade et al propose a decentralized data management system where all data access permissions are enforced using smart contracts and the audit trail of data access is stored in the blockchain, to support user data being shared among third-party entities. TrustChain is a platform for IoT device and data tracking and trading that aims to trace and manage data and devices without a central trusted authority.

However, the blockchain is computationally expensive and involves high bandwidth overhead and delays, which are unsuitable for most IoT devices. Our work is based on the existing research results and further explores a lightweight and efficient solution based on the blockchain to ensure the authenticity of identity, data, and behavior. The triple-trusting architecture not only solves the problem of trusted transmission of IoT data but also facilitates the innovation and development of the blockchain technology itself.

5 | CONCLUSIONS AND DIRECTIONS FOR FUTURE RESEARCH

Large-scale cooperation among smart entities of IoT, which has become a trend, requires frequent data flow among multiple stakeholders. The existing research has provided a good foundation for mass data transmission and interaction in IoT, eg, in multimedia IoT systems and WMSNs. However, sharing trusted data among trusted stakeholders in a trusted way still faces technical challenges, especially when the trusted third party cannot provide expected services. To alleviate this dilemma to some extent, this paper tries to present a SLTA that includes an oracle-based data collecting mechanism and the DID management mechanism. These concepts are used to ensure that the data collected from an IoT node cannot be modified, to ensure the credibility of the identity without a trusted third party, and to ensure trusted and decentralized data sharing, respectively. However, the blockchain has the characteristics of being computationally expensive and involves high bandwidth overhead and delays, whereas large-scale cooperation scenarios in IoT have the characteristics of dynamic changes in cooperative relationships, disturbance of participating nodes, elasticity of scale, and diversity and heterogeneity of nodes in computing, storage, communication, and other aspects. To make the blockchain applicable to IoT, we present a series of technological innovations that is also a part of key mechanisms of SLTA, including a new software-defined blockchain structure model, a lightweight Byzantine fault-tolerant algorithm, a low-overhead sequential storage model, etc.

ACKNOWLEDGEMENTS

The research is supported by the National Key R&D Program of China (grant 2016YFB1000100), by the National Natural Science Foundation of China (grant 61772030), and by the GF Innovative Research Program.

ORCID

Peichang Shi https://orcid.org/0000-0002-0973-7899
REFERENCES

1. Atzori L, Iera A, Morabito G. The Internet of Things: a survey. Computer Networks. 2010;54(15):2787-2805.

2. Whitmore A, Agarwal A, Xu L. The Internet of Things—a survey of topics and trends. Inf Syst Front. 2015;17(2):261-274.

3. Bader A, El Sawy H, Gharbieh M, Alouini M, Adinoyi A, Alshaalan F. First mile challenges for large-scale IoT. IEEE Commun Mag. 2017;55(3):138-144.

4. Alvi S, Afzal B, Shah G, Atzori L, Mahmood W. Internet of multimedia things: vision and challenges. Ad Hoc Netw. 2015;33:87-111.

5. Al-Turjman F, Radwan A. Data delivery in wireless multimedia sensor networks: challenging and defying in the IoT era. IEEE Wirel Commun. 2017;24(5):126-131.

6. Zheng Z, Xie S, Dai H, Chen X, Wang H. Blockchain challenges and opportunities: a survey. Int J Web Grid Serv. 2017;14(4):352-375.

7. Wang H, Shi P, Zhang Y. Jointcloud: a cross-cloud cooperation architecture for integrated internet service customization. Paper presented at: IEEE 37th International Conference on Distributed Computing Systems; 2017; Atlanta, GA. https://doi.org/10.1109/ICDCS.2017.237

8. Adler J, Berryhill R, Veneris A, Poulos Z, Veira N, Kastania A. ASTREA: a decentralized blockchain oracle. 2018. https://arxiv.org/pdf/1808.00528.pdf. Accessed August 1, 2018. arXiv:1808.00528v1.

9. Castro M, Liskov B. Practical Byzantine fault tolerance. In: Proceedings of the 3rd USENIX Symposium on Operating Systems Design and Implementation; 1999; New Orleans, LA.

10. Kotla R, Alvisi L, Dahlin M, Clement A, Wong E. Zyzzuya: speculative Byzantine fault tolerance. ACM SIGOPS Oper Syst Rev. 2007;41(6):45-58.

11. Nakamoto S. Bitcoin: a peer-to-peer electronic cash system. 2009. http://www.cryptovest.co.uk. Accessed October 31, 2008.

12. Lagutin D, Kortesniemi Y, Fotiou N. Enabling decentralised identifiers and verifiable credentials for constrained Internet-of-Things devices using OAuth-based delegation. Paper presented at: Workshop on Decentralized IoT Systems and Security; 2019; San Diego, CA.

13. Eyal I, Gencer A, Sirer E, Van Renesse R. Bitcoin-NG: a scalable blockchain protocol. Paper presented at: USENIX Symposium on Networked Systems Design and Implementation; 2016; Santa Clara, CA.

14. Baird L. The Swirlds Hashgraph Consensus Algorithm: Fair, Fast, Byzantine Fault Tolerance. Swirlds Technical Report SWIRLDS-TR-2016-01. College Station, TX: Swirlds Inc; 2016.

15. Luu L, Narayanan V, Zheng C, Baweja K, Gilbert S, Saxena P. A secure sharding protocol for open blockchains. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security; 2016; Vienna, Austria.

16. Ongaro D, Ousterhout J. In search of an understandable consensus algorithm. Paper presented at: USENIX Annual Technical Conference; 2014; Philadelphia, PA.

17. Gilad Y, Hemo R, Micali S, Vlachos G, Zeldovich N. Algorand: scaling Byzantine agreements for cryptocurrencies. In: Proceedings of the 26th Symposium on Operating Systems Principles; 2017; Shanghai, China.

18. Wood T, Singh R, Venkataaramani A, Shenoy P, Cechet E. ZZ and the art of practical BFT execution. In: Proceedings of the Sixth Conference on Computer Systems (EuroSys); 2011; Salzburg, Austria.

19. Conti M, Dehghanmehr A, Franke K, Watson S. Internet of Things security and forensics: challenges and opportunities. Future Gener Comput Syst. 2018;78(2):544-546.

20. Ambrosin M, Anzanpour A, Conti M, et al. On the feasibility of attribute-based encryption on Internet of Things devices. IEEE Micro. 2016;36(6):25-35.

21. Rahman H, Ahmed N, Hussain M. A hybrid data aggregation scheme for provisioning Quality of Service (QoS) in Internet of Things (IoT). Paper presented at: Cloudification of the Internet of Things; 2016; Paris, France. https://doi.org/10.1109/CİOT.2016.7872917

22. Hassanein H, Oteafy S. Big sensed data challenges in the Internet of Things. Paper presented at: International Conference on Distributed Computing in Sensor Systems; 2017; Santa Clara, CA.

23. Kumar M, Santhoshkumar S, Gowdhaman T, Shahajaan S. A survey on IoT performances in big data. Int J Comput Sci Mob Comput. 2017;6(10):26-34.

24. Taheerkordi A, Eliassen F. Data-centric IoT services provisioning in fog-cloud computing systems. In: Proceedings of the IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation; 2017; Pittsburgh, PA.

25. Long C, Cao Y, Jiang T, Zhang Q. Edge computing framework for cooperative video processing in multimedia IoT systems. IEEE Trans Multimedia. 2018;20(5):1126-1139.

26. Zarpelao B, Miani R, Kawakani C, de Alvarenga S. A survey of intrusion detection in Internet of Things. Inf Syst Front. 2015;17(2):261-274.

27. Dorri A, Kanhere S, Jurdak R. Towards an optimized blockchain for IoT. In: Proceedings of the Second International Conference on Internet-of-Things Design and Implementation; 2017; Pittsburgh, PA.

28. Ramachandran G, Krishnamachari B. Blockchain for the IoT: opportunities and challenges. 2018. https://arxiv.org/abs/1805.02818. Accessed May 8, 2018.

29. Khan M, Salah K. IoT security: review, blockchain solutions, and open challenges. Future Gener Comput Syst. 2017;82:395-411.

30. Dorri A, Kanhere S, Jurdak R, Gauravaram P. Blockchain for IoT security and privacy: the case study of a smart home. In: Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops); 2017; Kona, HI.

31. Choi S, Burm J, Sung W, Jang J, Reo Y. A blockchain-based secure IoT control scheme. Paper presented at: International Conference on Advances in Computing and Communication Engineering; 2018; Paris, France.

32. Zhu X, Badr Y, Pacheco J, Hariri S. Autonomic identity framework for the Internet of Things. Paper presented at: International Conference on Cloud and Autonomic Computing; 2017; Tucson, AZ.
33. Ayoade G, Karande V, Khan L, Hamlen K. Decentralized IoT data management using blockchain and trusted execution environment. Paper presented at: IEEE International Conference on Information Reuse and Integration; 2018; Salt Lake City, UT.

34. Yu B, Wright J, Nepal S, Zhu L, Liu J, Ranjan R. TrustChain: establishing trust in the IoT-based applications ecosystem using blockchain. *IEEE Cloud Comput.* 2018;5(4):12-23.

**How to cite this article:** Shi P, Wang H, Yang S, Chen C, Yang W. Blockchain-based trusted data sharing among trusted stakeholders in IoT. *Softw: Pract Exper.* 2021;51:2051–2064. [https://doi.org/10.1002/spe.2739](https://doi.org/10.1002/spe.2739)