Security in Edge Blockchains: Attacks and Countermeasures

CAO Yinfeng, CAO Jiannong, WANG Yuqin,
WANG Kaile, LIU Xun
(Department of Computing, The Hong Kong Polytechnic University,
Hong Kong SAR 852, China)

DOI: 10.12142/ZTECOM.202204002

https: //kns.cnki.net/kcms/detail/34.1294.TN.20221111.1503.004.html,
Published online: November 14, 2022

Manuscript received: 2022-09-16

Abstract: Edge blockchains, the blockchains running on edge computing infrastructures, have attracted a lot of attention in recent years. Thanks to data privacy, scalable computing resources, and distributed topology nature of edge computing, edge blockchains are considered promising solutions to facilitating future blockchain applications. However, edge blockchains face unique security issues caused by the deployment of vulnerable edge devices and networks, including supply chain attacks and insecure consensus offloading, which are mostly not well studied in previous literature. This paper is the first survey that discusses the attacks and countermeasures of edge blockchains. We first summarize the three-layer architecture of edge blockchains: blockchain management, blockchain consensus, and blockchain lightweight client. We then describe seven specific attacks on edge blockchain components and discuss the countermeasures. At last, we provide future research directions on securing edge blockchains. This survey will act as a guideline for researchers and developers to design and implement secure edge blockchains.

Keywords: blockchain; edge computing; security; survey

Citation (IEEE Format): Y. F. Cao, J. N. Cao, Y. Q. Wang, et al., "Security in edge blockchains: attacks and countermeasures," *ZTE Communications*, vol. 20, no. 4, pp. 3 – 14, Dec. 2022. doi: 10.12142/ZTECOM.202204002.

1 Introduction

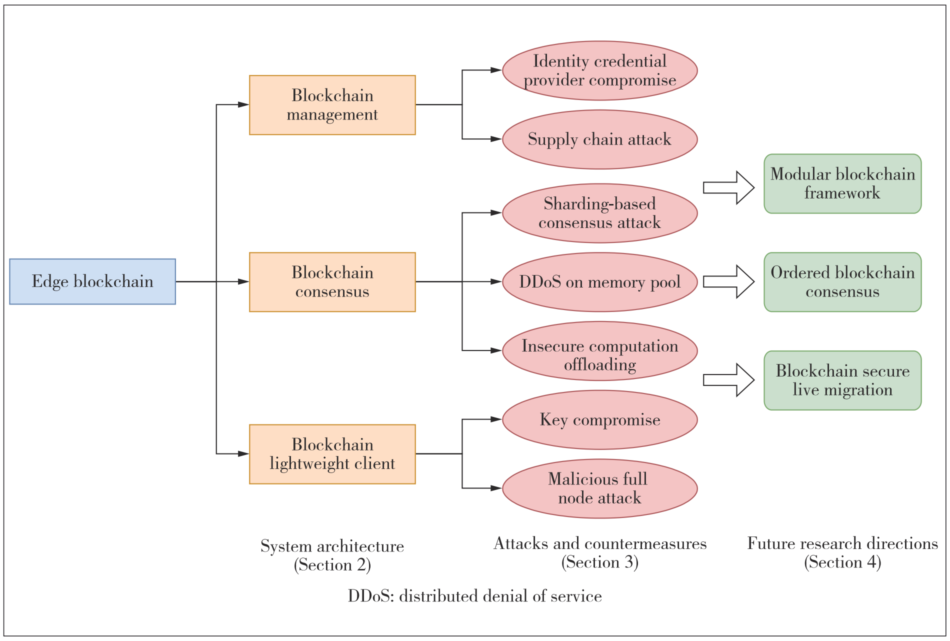
Edge computing has developed rapidly in recent years and raised wide interest from both industry and academia[1]. As a new computing model, edge computing extends cloud computing to the network edge and utilizes rich computation, storage, and networking resources on large-scale distributed devices. In edge computing, the optimization techniques on resource allocation and scheduling are extensively studied, enabling computation tasks to be divided and offloaded to the optimal edge devices according to different constraints. As a result, edge computing plays an important role in maintaining low latency, supporting heterogeneity, and improving applications' quality of service (QoS), such as virtual reality, distributed machine learning, wireless sensing, and robotics.

At the same time, the concept of edge blockchains has also been proposed. It refers to the blockchains deployed on edge computing infrastructures. Edge blockchains inherit favorable features from edge computing, like data privacy, scalable computing resources, and distributed topology. Thereby, edge blockchains are more suitable for large-scale applications than traditional blockchains hosted on the cloud or on-premise machines. For example, applications like blockchain-based federated learning, blockchain-based security middleware in the Internet of things (IoT), and metaverse essentially rely on edge blockchains[2 – 4].

However, the security in edge blockchains is not well understood in existing works. Specifically, in the works alleged "blockchain-based edge computing"[2, 5 – 6], the edge blockchains are typically assumed to be secure and trusted. On the contrary, in real implementation, it is challenging to protect and keep the edge blockchain networks functioning in edge environments for many practical reasons, like vulnerable low-end edge devices, unstable networks, and centralized provider corruption. Thus, it is desired to analyze the critical security issues facing edge blockchains.

To fill this gap, we investigate and evaluate the security in edge blockchains systematically. Our survey essentially differs from previous blockchain security surveys and provides more practical details[7 – 9]. As shown in Fig. 1, we start by describing the motivation and summarizing the system architecture and applications of edge blockchains to provide readers with a brief overview in Section 2. Then in Section 3, we discuss the core components of an edge blockchain, e.g., blockchain management, blockchain consensus, and blockchain

This work was supported by the Research Institute for Artificial Intelligence of Things, The Hong Kong Polytechnic University, HK RGC Collaborative Research Fund (CRF) under Grant No. C2004-21GF.


Edge blockchain
Blockchain management
Blockchain consensus
Blockchain lightweight client
Identity credential provider compromise
Supply chain attack
Sharding-based consensus attack
DDoS on memory pool
Insecure computation offloading
Key compromise
Malicious full node attack
Modular blockchain framework
Ordered blockchain consensus
Blockchain secure live migration
System architecture (Section 2)
Attacks and countermeasures (Section 3)
Future research directions (Section 4)
DDoS: distributed denial of service

▲**Figure 1. Structure of this survey**

lightweight clients, in terms of potential attacks and countermeasures. Finally, we point out the challenging issues and future directions for securing edge blockchains in Section 4.

2 Overview of Edge Blockchains

2.1 Motivations

In recent years, the blockchain technology and its applications have received extensive attention from the research community and industry[10]. Blockchain is a decentralized ledger-based Byzantine fault tolerant (BFT) consensus system. Under a bounded number of adversary environments, blockchain nodes can reach chain-linked agreements on incoming transactions with traceability, immutability, and transparency. Besides, the consensus procedure does not rely on a trusted third party (TTP), making blockchain systems trustless and hard to tamper with.

Nowadays, the blockchain technology is still facing several bottlenecks, thus seriously restricting its application scenarios and making it inaccessible in the real world. Among them, blockchain's decentralization, scalability, and security are considered the most significant and recognized as a trilemma[11]. Generally speaking, existing works cannot well satisfy all three properties together.

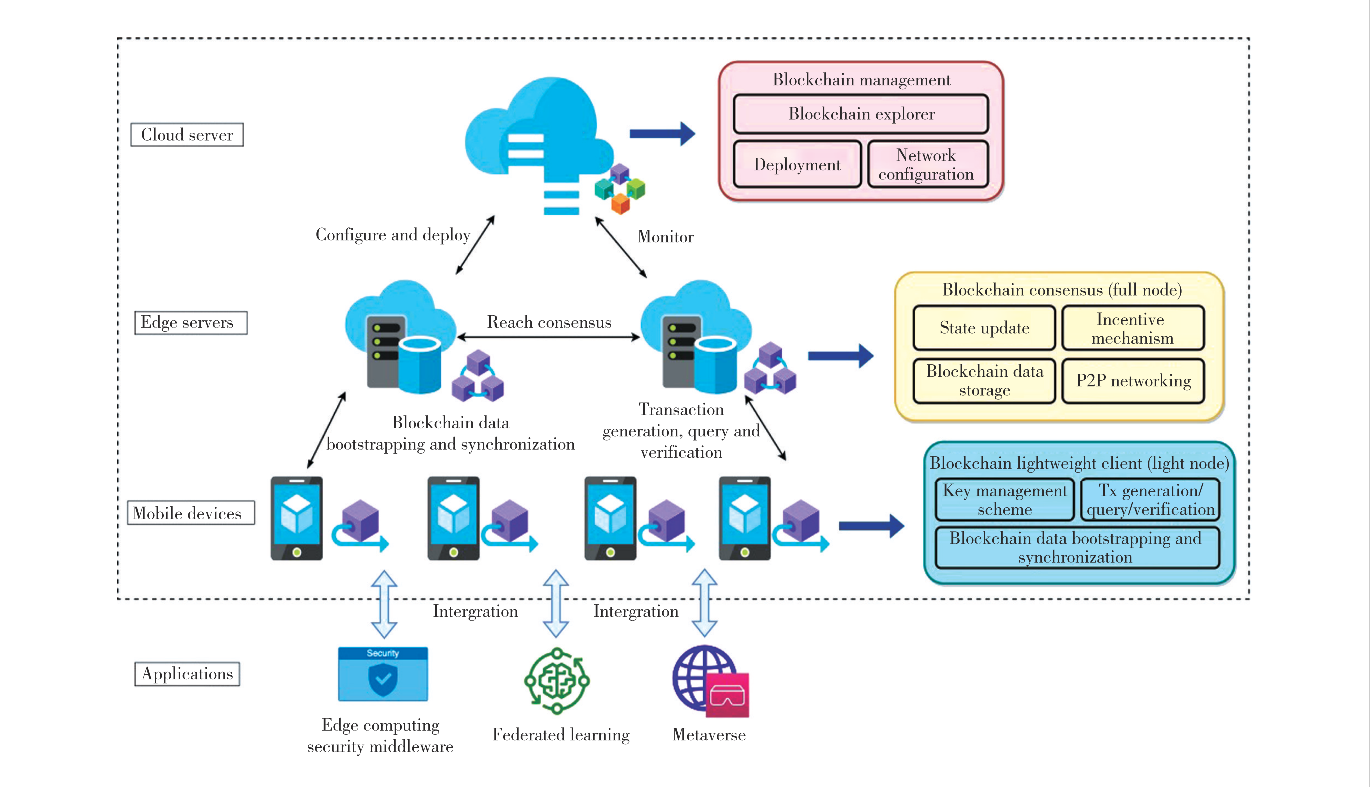
- Decentralization. Making blockchain run without trust depends on a small group of centralized actors with specialized rights.
- Scalability. Processing numerous transactions in the network simultaneously with low latency.
- Security. Resisting a certain percentage of Byzantine nodes that can conduct arbitrary adversary behaviors.

Although blockchain theoretical advancement keeps appearing, some researchers have begun to focus on infrastructure-level solutions to improving blockchain performance. Edge computing shares a similar system architecture with blockchain and can provide the needed computing resources for blockchain systems, which can be a promising option[12 – 13]. By employing edge computing as blockchain infrastructures to realize edge blockchains, blockchain trilemma can be further resolved simultaneously in terms of the above-mentioned properties. In particular, 1) for keeping decentralization, edge computing physically guarantees the hierarchical and decentralized architecture of blockchain. Edge computing has a layered architecture with a large-scale distributed edge device network to keep the on-device blockchain nodes from centralization. 2) For improving scalability, edge computing has rich computation, storage, and networking resources accomplished with automatic optimization of resource allocation and scheduling. These resources can be utilized by blockchain to realize a large-scale blockchain network. 3) For enhancing security, edge computing provides permission environments with data privacy guarantees, reducing Byzantine nodes' risks and thus relaxing the blockchain security assumption for better performance. In conclusion, edge blockchains can offer better decentralization, scalability, and security with lower latency for applications than normal blockchains deployed on public cloud environments or distributed individual devices.

2.2 Architecture

To comprehensively analyze and understand edge blockchains, we first present their architecture, components, and functionalities. Based on the existing literature and platforms, we find that the edge blockchain system architecture typically follows a three-layer pattern including a cloud server layer, an edge server layer, and a mobile device layer, with different blockchain components and functionalities, as shown in Fig. 2. To be specific, we conclude each layer's components and functionalities as follows:

- Cloud server layer: blockchain network management. The cloud server in edge computing has knowledge of network specifications and attached edge devices connectivity. Therefore, the cloud servers are typically set to configure, deploy, and monitor the edge blockchain networks to reduce management costs.



▲Figure 2. System architecture of a typical edge blockchain which follows a three-layer pattern with different blockchain components and functionalities

- Edge server layer: blockchain consensus for incoming transactions. Edge servers are close to the data source and provide more computing resources than mobile devices. Thus, it is reasonable to deploy blockchain full nodes at this layer to have sufficient resource support for updating ledger states and incentive nodes, storing blockchain data, and communicating with other nodes.
- Mobile device layer: blockchain lightweight client (light node) for transaction operations (generation, query, and verification). Edge applications interact with blockchain through numerous transactions from large-scale mobile devices. Deploying the interfaces for transaction operations and maintaining partial blockchain data (blockchain data bootstrapping and synchronization) at this layer can significantly reduce the latency and improve QoS. Besides, the key management schemes for protecting signing keys are also integrated to support transaction operations.

2.3 Applications

Edge blockchains feature high QoS and security guarantees in edge environments. In the current stage, edge blockchain solutions are application-specific, which means that they are typically embedded with applications to improve their performance. Here we summarize the three representative types as follows.

- Security middleware for edge computing[2, 14 – 15]. Edge blockchains can be utilized as security middleware to tackle security issues in the edge computing infrastructure. For instance, detecting unstable or low-performance edge devices and designing strategies to avoid using these devices are challenging research issues in the edge resources optimization area. To address these issues, reputation systems with incentive mechanisms can be built upon edge blockchains. They record the status of edge devices and provide trusted reference information for strategy design and decision making in edge optimization algorithms. Besides, other efforts like secure data sharing methods, authentication schemes, and control systems based on edge blockchains are also proposed to enhance edge computing security.
- Edge-based federated learning[3, 16 – 17]. Edge-based federated learning is a distributed machine learning scheme that collects closed-source data to train global models in a privacy-preserving and personalized manner. However, due to the self-voluntary ways to contribute to model updates, malicious behaviors may occur and affect the quality of global models, e.g., poison attacks. To this end, blockchain is proposed to provide failure tolerance ability, malicious behavior detection, and incentive mechanisms for securing and boosting federated learning.
- Metaverse[4, 18 – 19]. Metaverse is a trendy edge application aiming to build a virtual world with immersive experience. Edge-based VR and blockchain-based economic systems are

two critical techniques for the metaverse. Edge blockchains provide lower latency, better decentralization, and better personal data privacy than blockchains in the cloud, thus making the metaverse scalable and trusted.

2.4 Challenges

Deploying blockchains at the edge will bring extra challenges, especially from security aspects. On the one hand, vulnerable edge devices, unstable network conditions, and physical accessibility expose many attack interfaces on edge blockchains to adversaries. On the other hand, designing sufficient and efficient security solutions on resource-constrained edge devices is challenging.

For example, efficient and secure key management is challenging in edge blockchains. Traditional methods like using custodial wallet software require considerable computing resources on edge devices, which are also insecure since external attackers can access the devices. Ideal solutions should be lightweight but also can prevent such kinds of attacks. Another example can be the task-offloading feature of edge computing. Offloading tasks to arbitrary nodes in blockchain networks is risky since the blockchain nodes do not trust each other. Malicious nodes could collect the offloaded tasks to gain illegal benefits and launch attacks by forging identities.

3 Attacks and Countermeasures

In this section, we describe the critical security issues and attacks of edge blockchains in each layer. We also present and analyze the state-of-the-art countermeasures for reference in each subsection. We summarize these contents with brief descriptions in Table 1.

3.1 Blockchain Management

Blockchain management aims to configure, deploy, and monitor edge blockchain networks. In edge blockchains, such procedures are typically implemented by centralized service providers, e.g., Blockchain-as-a-Service (BaaS) platforms, due to cost-effective concerns[20-23]. These platforms provide the tools or software development kits (SDKs) to define the blockchain network in client software, access control, deployment methods, etc. For example, AWS Blockchain Template is a tool for configuring cloud-based Ethereum[24] or hyperledger fabric networks[25].

3.1.1 Identity Credential Provider Compromise

In edge blockchains or other consortium blockchains, identity credentials are required to authenticate the participation legality of users or organizations. Identity credentials can be certificate authority/public key infrastructure (CA/PKI) certificates and public/private key pairs, which are generated and assigned to blockchain nodes. These credentials specify the vote right, communication channels, and data access. For popular frameworks in edge blockchains, like Hyperledger Fabric, X.509 CA-based Membership Service Provider (MSP) is responsible for participation identity management; in IBM blockchain, blockchain identities are associated with Azure Active Directory, a unified access control mechanism in Azure Cloud[26]. In edge blockchain literature, similar mechanisms are also applied to authenticate edge devices that run blockchain nodes[27 - 29]. However, due to the centralized nature of this procedure, blockchain management procedures in edge blockchains are vulnerable to many attacks, even to traditional cyber attacks.

Although nodes themselves keep the credentials, the issue, update, and revoke operations are typically performed by centralized providers (e.g., blockchains using CA/PKI), which is risky to adversaries. Existing works show that if such providers are compromised, many other level attacks may be conducted and further damage the blockchain networks[30 - 31]. Malicious providers can manipulate and subvert identity management by making legal credentials invalid, refusing to issue, and even issuing illegal credentials to launch a Sybil attack. Eventually, malicious providers will control the full blockchain networks and could launch arbitrary attacks.

State-of-the-Art countermeasures focus on making blockchain identity management decentralized and transparent. In Geth (Proof of Authority consensus mode) and Tendermint, new validators are elected to have vote rights by original validators, which are initially from the hard-coded genesis block[32 - 33]. This way increases the difficulty for adversaries to compromise since it is equivalent to tamper the entire blockchain. The substantial verification, update, and revocation operations are also on-chain. Some works extend similar ideas and construct new identity blockchains, which are specifically designed for managing identities on other blockchains[34 - 36].

▼Table 1. Attacks and countermeasures on edge blockchain components

Components	Attacks	Countermeasures	Related Works
Blockchain management	Identity credential provider compromise	Decentralization and transparent identity management	Refs. [30 - 36]
	Supply chain attack	Threat detection system and automated code analysis	Refs. [39 - 48]
Blockchain consensus	Sharding-based consensus attack	Atomic commit and order-fairness consensus	Refs. [59, 62 - 65]
	DDoS on a memory pool	Increase of the costs of malicious transactions	Refs. [66 - 72]
	Insecure computation offloading	Secure multi-party computation	Refs. [13, 76 - 79]
Blockchain lightweight client	Key compromise	New recovery operations on blockchain and robust key management	Refs. [86 - 92]
	Malicious full node	Reputation system and game-theoretic approach	Refs. [82, 93, 95]

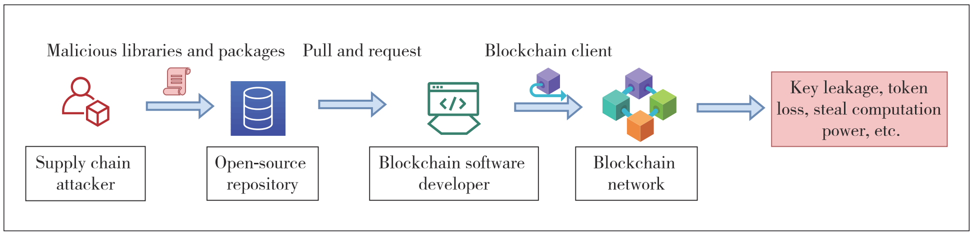
DDoS: distributed denial of service

3.1.2 Supply Chain Attack

In practice, blockchain nodes are implemented by blockchain client software like Geth[32] and Bitcoin Core[37]. These blockchain clients are developed or orchestrated from multiple libraries, packages, and dependencies, providing consensus, blockchain data storage, APIs, wallet functionalities, etc. Due to the nature of decentralization and trust concerns, their blockchain components are usually supplied by open-source projects. For example, Geth involves Web3.js library to provide APIs for blockchain, and smart contract interactions[38].

A supply chain attack (e.g., a third-party attack, a value-chain attack, or a backdoor breach) aims to inject malware or malicious hardware by hiding in upstream supplied system components to damage software. Historic attacks were mainly launched by suppliers in traditional information and communications (ICT) technology areas. However, recent accidents show that it can also affect blockchain since blockchain projects are mostly built by open-source dependencies to increase transparency. As shown in Fig. 3, attackers may upload predesigned malicious libraries and packages to open-source repositories by compromising blockchain managers, and then deliver them to blockchain software developers. Users will be compromised when they run crafted blockchain software like wallets[39]. Likewise, there is so-called mining malware that pretends to be normal browser plugins, executable programs, and miner tools, stealing the computation power of devices to obtain benefits[40]. In edge blockchains, such attacks are noteworthy since the blockchain clients running on edge devices are provided and maintained in a similar way. Even worse, edge blockchain networks are dynamic, and edge devices frequently join and leave the networks by installing the blockchain client software from different sources. These processes expose additional attack interfaces for supply chain attacks.

The preventive solutions try to eliminate the risks from both the upstream components supplier side and the device side[41 – 43]. On the one hand, researchers and developer communities use various security mechanisms to assert the projects hosted in open-source repositories. Many scoring and threat detection and analysis systems like OpenSSF Metrics and OpenSSF Scorecard are built to provide an overview of the security status for developer reference[44 – 45]. They calculate the scores according to the code maintenance status, vulnerability existence, and programming specification as metrics. On the other hand, the automated code analysis project, and services for detecting blockchain software and smart contracts are emerging[46 – 48]. They can check sensitive codes and functions like money transfer, deploying contracts, and making signatures by semantics formalization. This way is more active than the former but may bring huge additional development costs.


Malicious libraries and packages
Pull and request
Blockchain client
Supply chain attacker
Open-source repository
Blockchain software developer
Blockchain network
Key leakage, token loss, steal computation power, etc.

▲ **Figure 3. Supply chain attacks in blockchain: attackers can inject malicious scripts into libraries and packages to damage blockchain networks**

3.2 Blockchain Consensus

Consensus is a core component of blockchain systems that refers to the continuous agreement protocol on blocks/transactions among multiple blockchain nodes. Blockchain consensus can reach an agreement and update node states under the existence of Byzantine nodes. Byzantine nodes can behave arbitrarily to achieve malicious targets except by breaking cryptography primitives, and they can also cooperate. For example, Byzantine nodes can keep silent to pretend to crash or corporately send fake messages to foolish honest nodes. Currently, there are mainly two types of blockchain consensus: the Nakamoto style and the traditional BFT style. Nakamoto style consensus includes Proof of Work (PoW), Proof of Stake (PoS), Proof of Authority (PoA)[37, 49, 32], etc, which rely on external validity rules like mining power, stocks, and authority to reach agreements. BFT style consensus purely concerns the votes on broadcasted values, like practical Byzantine fault tolerance (PBFT), HotStuff, and Honey badger[50 – 52]. Besides, blockchain consensus is also highly related to hardware, blockchain data structure, networking algorithms, and blockchain lightweight client design[53 – 54].

Consensus is a vulnerable component due to the complexity and non-deterministic procedures. Existing attacks focus on breaking two consensus features as follows:

- Consistency (safety): If any two honest nodes in the blockchain network maintain two blockchains, they should be on the same chain.
- Liveness: If the honest nodes receive a transaction, the transaction should be included in all blockchains maintained by honest nodes after the consensus procedure.

Literally speaking, if the attack breaks consistency, there will be unexpected blockchain forks or double spending events. If the attack breaks liveness, the consensus will halt and no agreement has been reached for incoming transactions.

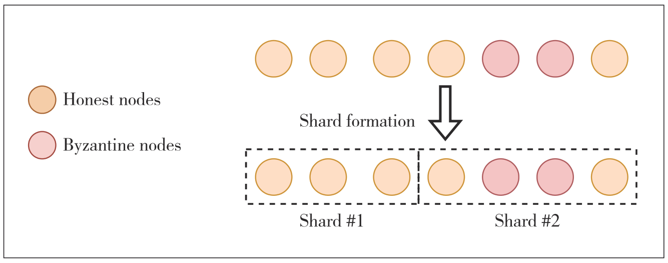
3.2.1 Sharding-Based Consensus Attack

In edge blockchain networks, the numerous edge devices require the blockchain consensus to be scalable to maintain high Transaction per Second (TPS). However, the theoretical limitations make the communication complexity hard to be sub-quadratic (BFT style consensus). Sharding is a celebrated and preferred technique to deal with scalability issues in edge computing[55 – 57]. Generally speaking, sharding splits the blockchain networks into several pieces, where each piece individually deals with transaction consensus and data storage. This way can reduce the communication to nearly linear as well as the storage cost[58 – 59].

However, in practice, the transaction may be related to multiple shards, which brings extra security issues. For example, in the unspent transaction output (UTXO) model, the outputs of a transaction must be equal to (or smaller than) its input. If the inputs come from different shards, nodes in one shard cannot verify the validity of this transaction for they have no knowledge of other shards. In the account/balance model, the transaction is still probably from different shards when the shard number increases. Thus, a cross-shard consensus protocol as the coordinator is needed to deal with this situation. The typical solution is using the atomic commit (AC) protocol to implement this coordinator. However, existing works show that existing coordinators are vulnerable to various attacks, including transaction forging attacks, message withholding attacks, publish-revert attacks, and replay attacks, which can break the liveness and safety of sharding-based consensus[60 – 61]. Generally speaking, these attacks leverage the knowledge difference and message delay during the AC process, further cheating or isolating the honest shard chains.

Another fundamental security issue of sharding-based consensus is committee selection for shard formation. Traditional blockchain consensus assumes that the number of Byzantine nodes is under a certain percentage (security boundary), e.g., 50% for PoW and 33% for PBFT. When the blockchain network is splitted into reveal shards, the number of Byzantine nodes may exceed security boundaries in particular single shards. This issue is also called a signal shard takeover attack. For example, a blockchain network runs PBFT with seven nodes, where five nodes are honest and two nodes are Byzantine. If the network manager randomly selects committees in shards as shown in Fig.4, the second shard will be compromised since there are more than 33% Byzantine nodes. This situation comes from the uncertainty of Byzantine nodes, making managers hard to decide which node is honest.

For the first security issue, researchers try to design new AC protocols that have extra features like termination. It requires all involved shards on a cross-shard transaction to eventually decide on it. Besides, in real implementation, a "garbage collection" is used for dealing with uncompleted cross-shard transactions[62 – 63]. However, such works are specific to their blockchain systems, and cannot be directly applied to other blockchains. Traditional non-blocking atomic commit (NC-AC) is also needed to be significantly modified to be compatible with the blockchain system[64]. For the second security issue, the public verifiable randomness sources and countermeasures for active adversaries are introduced. The randomness sources provide the reference for shard formation. Being unpredictable and uniform can minimize the probability of selecting excessive Byzantine nodes in shards[59]. For active adversaries, which corrupt nodes after shard formation, there are also mechanisms to limit their abilities of malicious voting[65].



▲ **Figure 4. Signal shard takeover attack: shard may contain exceeded numbers of Byzantine nodes after committee selection**

3.2.2 DDoS on Memory Pool

A memory pool in a blockchain system is a caching area for receiving, verifying, and ranking incoming transactions before consensus. The memory pool is the first step for processing transactions. Thus its performance will be the bottleneck of TPS. For example, in Bitcoin, the miner first checks the validity of transactions in terms of signatures, UTXOs, formats, etc. Then the transactions will be put in a memory pool waiting to be mined into blocks[37]. The ranking of transactions depends on the mining fee attached to the transactions. High mining fees stimulate miners to mine transactions in a high rank, making them early confirmed. Besides, the relay fees are also required for miners relaying the transactions to each other. Other blockchain systems are designed with similar philosophies. The differences lay in the requirement for fees. In edge blockchains, the fees are omitted and the ranking is decided by the arriving time or other parameters[66 – 68].

Recent studies show that the DDoS attack can significantly affect the memory pool, prohibiting normal transactions from being confirmed[69 – 72]. Attackers first allocate multiple Sybil accounts with enough balances for paying transaction fees and relay fees. Then they initiate a large number of unconfirmed transactions that transfer money to each other to several blockchain nodes in a short time period. When the transaction arrival rate is larger than the confirmation rate of blockchain consensus, there will be a transaction backlog, and the blockchain nodes have to increase the size of memory pools eventually. Although the consensus processes as normal, the actual TPS for normal transactions will be decreased. Attackers try to maximize the number of these transactions in the memory pool but do not want them to be confirmed since it will cost more fees. Therefore, these transactions typically only have relay fees to reduce the attack costs. In edge blockchains, conducting such attacks is more possible than doing this in cryptocurrency. The reasons include that the transactions in edge blockchains are application-specific and may not need to pay money, and the corrupted edge devices can easily generate a large number of transactions.

Existing solutions focus on increasing the costs of launching such attacks to further prevent them from happening. Researchers set additional constraints to filter the transactions that are likely to be malicious. The constraints consider whether the parents transactions are confirmed previously and therefore pay mining fees[69 – 72], or set the relay fees dynamically increasing when the memory pool size is too large[70]. These solutions only care about cryptocurrency systems, but such mechanisms may not be feasible in edge blockchain networks.

3.2.3 Insecure Computation Offloading

Computation offloading is a unique technique in edge computing. It transfers resource-intensive computational tasks to other nearby devices by dividing and optimizing tasks. In this way, resource-constrained devices reduce the burden and are capable of dealing with complex tasks. Computation offloading is extensively studied and applied in edge computing, and many edge applications essentially rely on it, such as distributed machine learning, video surveillance, and VR/AR[73 – 75]. In edge blockchains, offloading is also utilized for reducing the blockchain consensus costs on mobile devices[76 – 77, 13]. Researchers model the consensus tasks and edge compute services pricing as Stackelberg games to improve the system throughput and optimize the accessibility of the blockchain network.

However, such offloading methods cannot well meet the security requirements of blockchain. Even though the consensus tolerates a certain percentage of Byzantine nodes, malicious edge computing service providers (e.g., corrupted edge servers) are still possible to break the threshold. Specifically, malicious providers can execute other nodes' consensus tasks and act like them simultaneously. This behavior is equivalent to corrupting honest nodes in the blockchain since malicious providers obtain free computation power paid by honest nodes, which is definitely out of the BFT model definition. Consequently, the percentage of Byzantine nodes in blockchain networks will increase and finally become overwhelming.

Secure multiparty computation (SMPC) and outsourced computing can be promising solutions to addressing these issues. SMPC is a cryptographic technique that enables multiple parties to jointly compute tasks without revealing their own private inputs and outputs[78]. With the development advancing, its efficiency is becoming acceptable for edge and IoT devices. Combining the SMPC with blockchain and offloading can prevent malicious computing service providers from manipulating outsourced blockchain tasks[79].

3.3 Blockchain Lightweight Client

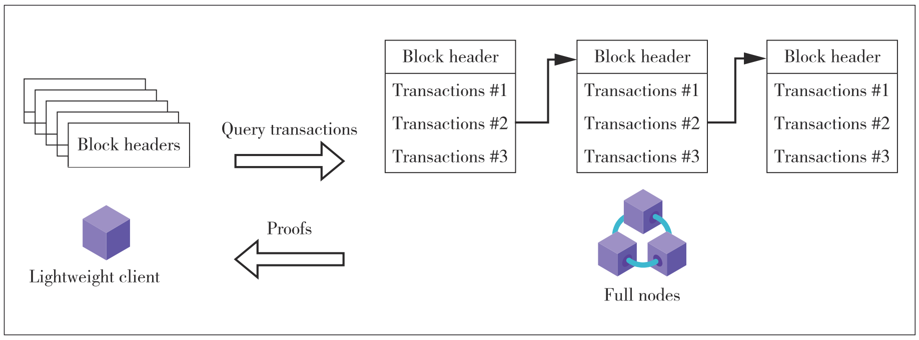
Blockchain lightweight client is another critical building block of a blockchain system, especially for developing edge blockchains. It contains transaction generation, query, and verification schemes with blockchain data bootstrapping and synchronization procedures. In practice, a blockchain lightweight client typically does not directly participate in consensus like blockchain full nodes do to save computation, storage, and networking resources. Therefore, a blockchain lightweight client is suitable to be integrated into mobile applications and run on resource-constrained devices in edge networks[80, 12, 81]. Specifically, a lightweight client contains the following functionalities[82]:

- Bootstrapping and synchronization: Given a blockchain genesis block or file, the client should synchronize all the state metadata from full nodes (e.g., all block headers) with bootstrapping proofs. When the full nodes update their states (e.g., new blocks), the client should also synchronize it and update state metadata with synchronization proofs.
- Transaction generation, query, and verification: Clients should generate valid transactions for full node updating its states. After that, a client can query the existence of submitted transactions confirmed in blockchain states and verify the result proofs.

Bitcoin simplified payment verification (SPV) is believed to be the first implementation of a lightweight client[37]. As shown in Fig. 5, it only stores the block headers of the longest chain locally, which is initially downloaded and periodically synchronized from nearby full nodes. Upon receiving transaction verification requests, lightweight clients retrieve the blocks that contain the transactions with corresponding Merkle branches for verifying their existence.

3.3.1 Key Compromise

Key management refers to the schemes of generating, updating, using, and deleting cryptographic keys. In the blockchain context, the keys are employed for identifying edge nodes, signing transactions, and encrypting data. Due to the decentralized nature, such keys are usually kept by the user sides, and no managers are responsible for them. In well-established blockchain wallets like MetaMask[83], the key files are stored locally with mnemonic phrase encoding and password/biometric authentication protection. Users need to input the correct password or biometric information to unlock the key inside the



▲ **Figure 5. Bitcoin simplified payment verification (SPV): a lightweight blockchain client for Bitcoin, which only stores the block headers to reduce costs** wallet to sign transactions.

Although many elaborately crafted key management schemes are designed and implemented for high security and usability, the keys are still extremely vulnerable to software bugs, hardware failure, and even simple human errors[84 – 85]. Key compromise is still a significant security issue that remains unsolved. To add insult to injury, these issues are more likely to happen in edge blockchains since edge devices (like IoT devices) are mostly low-end in hardware and software with few sufficient security mechanism. Besides, the devices are also physically accessible and controllable. Attackers can attack the devices through various interfaces existing on edge devices and applications. As a result, the keys managed in edge devices are highly risky of being theft, lost, and broken.

Existing efforts to enhance the security and usability of blockchain key management are twofold. First, from the blockchain side, many schemes are proposed to replace or supplement the transaction verification in blockchains to realize key compromise protection[86 – 89]. The general idea is to allow new operations to claim new keys or recall transactions for users who are theft or lose their keys accidentally. This way serves as remedial measures for unlucky users but creates additional difficulties and lowers the blockchain TPS for normal users. From the device side, advanced cryptography primitives are applied to minimize the risk of key compromise. Group signature, threshold signature, and hierarchical key derivation are used to construct robust key management schemes[90 – 92]. These schemes can provide additional rescue solutions, present informative network typology, and set flexible access control in edge blockchain key management.

3.3.2 Malicious Full Node Attack

The purpose of a malicious full node is to influence light nodes that interact with the blockchain network via a light protocol and inject adversarial blocks. Light clients have poor bandwidth and limited storage capacity. To improve the efficiency of light clients, firstly, they do not store complete ledger information; secondly, they generally verify the validity of the chain within a limited scope. For example, in SPV, light clients validate the chain only through block headers and request Merkle from full nodes on demand to verify that a specific transaction is valid. Incomplete validation makes it possible for malicious full nodes to inject adversary chains into light clients.

Malicious full nodes can create forks in the blockchain. A fork consists of blocks with block headers that satisfy block header validation and adversarial status. Honest full nodes will immediately reject these adversary blocks because they fail in state validation. However, since light clients can only perform header validation, but not state validation, the fork is also a normal chain from the perspective of light clients. If the adversary chain contains more work than the honest chain, according to the longest chain rule, a light client will accept the adversary chain. In addition, a patient adversary willing to wait (days or months) can obtain a high probability of successfully injecting a forged state into a light client. Considering a node, such as an IoT node with a limited battery, is operating in a duty cycle mode and periodically active, the longer the interval between two active states, the higher the adversary probability of successful state injection. Moreover, the adversary also has the probability of having a successful adversarial chain at any random point in time, so it may also successfully convince a light node[93].

Existing solutions focus on building reputation systems and using game-theoretic approaches to secure the light client from malicious full nodes. In reputation systems, miners are ranked by their consensus contribution[94]. Light clients cache the recent blocks from miners with good reputations to securely download blockchain data. Game-theoretic approaches use smart contracts as a trusted arbiter to deal with the client and a set of full nodes[95]. Participants need to deposit some funds on the arbiter contract as collateral. Malicious behaviors like sending fake blocks will be plenty by costing the deposited funds, thus encouraging the full node to provide block data honestly.

4 Future Research Directions

In this section, we point out some specific future research directions that are related to the security of edge blockchains. We envision these directions being significant in future edge blockchains.

4.1 Modular Blockchain Framework

Existing frameworks in edge blockchains only support one or multiple fixed components such as consensus algorithms, databases, and communication protocols. This fact significantly decreases the resistance to supply chain attacks and amplifies the attack revenue. Imagine if some widely used consensus algorithms or other components are suddenly found vulnerable, all the blockchain systems with those components will be risky and hard to be fixed in a short time. This is because the interfaces, data structures, and algorithms among these components are highly coherent. Developers do not have much flexibility to adjust them when security issues occur. Besides, the valuable on-chain assets and data make blockchain systems hard to be readily updated as normal software.

In our previous work, PolyChain proposes a modular blockchain framework, where the main components are fully pluggable and changeable[96]. We divide the blockchain into four components: application component, consensus component, storage component, and network component. This provides much flexibility when facing attacks. In PolyChain, developers can replace vulnerable components with low costs to avoid potential damage. Other works with similar philosophy also emerged recently, and better solutions to blockchain modularization remain to be explored[97 – 98].

4.2 Ordered Blockchain Consensus

Some transactions have inherent relationships and dependencies in certain applications. For example, in Decentralized Finance (DeFi) and Central Bank Digital Currencies (CBDC) [99 – 100], manipulating transaction confirmation orders can launch financial attacks on smart contracts[101]. This is because when users need to submit a batch of transactions to these applications, their exact confirmation order is not guaranteed in most existing consensus algorithms. Miners usually include transactions in blocks according to the attached fees. Many blockchain financial infrastructures are also deployed in edge environments, such as Bitcoin ATMs and cryptocurrency wallets. Therefore, we need to guarantee the security of transaction confirmation orders in edge blockchains.

One easy and safe way is submitting transactions to consensus one by one and waiting for confirmation, but this is inefficient when there are a large number of pending transactions. New efforts on consensus need to provide security guarantees on transaction orders while keeping high efficiency. This is challenging since it basically requires designing extra consensus rounds expressly agreeing on transaction orders.

4.3 Blockchain Secure Live Migration

In edge blockchains, multiple small-size blockchain networks may exist for specific user groups and applications. However, accessing these size-constrained blockchains can be difficult for they are only deployed in a limited number of edge devices with poor network connectivity. Simply scaling the blockchain network by setting up new nodes is a naive solution, but this will occupy other device computing resources and interrupt blockchain consensus, causing further security concerns for the blockchain network, like congestion.

Live migration is a technique that transfers services or processes across computing infrastructures without disrupting normal operations. It has been extensively studied in cloud computing for load balancing, resource management, server consolidation, predictive maintenance, and QoS improvement[102]. Such a technique is also beneficial to edge blockchains. It can reduce the latency of accessing the above-mentioned size-constrained blockchain with low costs and high QoS. Existing migration techniques only focus on container or process architecture, which may not perform well on blockchain since they do not consider the specific architecture of blockchain systems. More effective and secure solutions can be adopted by separately migrating different components of the blockchain system, such as blockchain data and memory pool transactions while keeping consensus running for security[103].

5 Conclusions

Integrating blockchain with edge computing is a valuable landscape in future wireless communication. Many efforts have been made to make blockchain securely run in adversarial environment. However, the features of edge computing bring new security issues, which have not been extensively studied and addressed in previous literature. Many attacks on edge blockchains components are not well understood and prevented. Through this study, we comprehensively review the security of edge blockchains in terms of attacks, countermeasures, and future directions. We envision this survey acting as a security guideline for designing and developing edge blockchains.

References

[1] CAO K Y, LIU Y F, MENG G J, et al. An overview on edge computing research [J]. IEEE access, 8: 85714 – 85728. DOI: 10.1109/ACCESS.2020.2991734

[2] HE Y, WANG Y H, QIU C, et al. Blockchain-based edge computing resource allocation in IoT: a deep reinforcement learning approach [J]. IEEE Internet of Things journal, 2021, 8(4): 2226 – 2237. DOI: 10.1109/JIOT.2020.3035437

[3] LU Y L, HUANG X H, DAI Y Y, et al. Blockchain and federated learning for privacy-preserved data sharing in industrial IoT [J]. IEEE transactions on industrial informatics, 2020, 16(6): 4177 – 4186. DOI: 10.1109/TII.2019.2942190

[4] XU M R, NIYATO D, KANG J W, et al. Wireless edge-empowered metaverse: a learning-based incentive mechanism for virtual reality [C]//IEEE International Conference on Communications. IEEE, 2022: 5220 – 5225. DOI: 10.1109/ICC45855.2022.9838736

[5] SHENG H, WANG S, ZHANG Y, et al. Near-online tracking with co-occurrence constraints in blockchain-based edge computing [J]. IEEE Internet of Things journal, 2021, 8(4): 2193 – 2207. DOI: 10.1109/JIOT.2020.3035415

[6] RAHMAN M A, HOSSAIN M S, LOUKAS G, et al. Blockchain-based mobile edge computing framework for secure therapy applications [J]. IEEE access, 2018, 6: 72469 – 72478

[7] LI X Q, JIANG P, CHEN T, et al. A survey on the security of blockchain systems [J]. Future generation computer systems, 2020, 107: 841 – 853. DOI: 10.1016/j.future.2017.08.020

[8] TAYLOR P J, DARGAHI T, DEHGHANTANHA A, et al. A systematic literature review of blockchain cyber security [J]. Digital communications and networks, 2020, 6(2): 147 – 156. DOI: 10.1016/j.dcan.2019.01.005

[9] ZHANG R, XUE R, LIU L. Security and privacy on blockchain [J]. ACM computing surveys, 2020, 52(3): 1 – 34. DOI: 10.1145/3316481

[10] MONRAT A A, SCHELÉN O, ANDERSSON K. A survey of blockchain from the perspectives of applications, challenges, and opportunities [J]. IEEE access, 7: 117134 – 117151. DOI: 10.1109/ACCESS.2019.2936094

[11] ZHOU Q H, HUANG H W, ZHENG Z B, et al. Solutions to scalability of blockchain: a survey [J]. IEEE access, 8: 16440 – 16455. DOI: 10.1109/ACCESS.020.2967218

[12] YANG R Z, YU F R, SI P B, et al. Integrated blockchain and edge computing systems: a survey, some research issues and challenges [J]. IEEE communications surveys & tutorials, 2019, 21(2): 1508 – 1532. DOI: 10.1109/COMST.2019.2894727

[13] XIONG Z H, ZHANG Y, NIYATO D, et al. When mobile blockchain meets edge computing [J]. IEEE communications magazine, 2018, 56(8): 33 – 39. DOI: 10.1109/MCOM.2018.1701095

[14] MA Z F, WANG X C, JAIN D K, et al. A blockchain-based trusted data management scheme in edge computing [J]. IEEE transactions on industrial informatics, 2020, 16(3): 2013 – 2021. DOI: 10.1109/TII.2019.2933482

[15] KANG J W, YU R, HUANG X M, et al. Blockchain for secure and efficient data sharing in vehicular edge computing and networks [J]. IEEE Internet of Things journal, 2019, 6(3): 4660 – 4670. DOI: 10.1109/JIOT.2018.2875542

[16] NGUYEN D C, DING M, PHAM Q V, et al. Federated learning meets blockchain in edge computing: opportunities and challenges [J]. IEEE Internet of

Things journal, 2021, 8(16): 12806 – 12825. DOI: 10.1109/jiot.2021.3072611

[17] MAJEED U, HONG C S. FLchain: federated learning via MEC-enabled blockchain network [C]//Proceedings of 2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS). IEEE, 2019: 1 – 4. DOI: 10.23919/apnoms.2019.8892848

[18] KIM W S. Edge computing server deployment technique for cloud VR-based multi-user metaverse content [J]. Journal of Korea multimedia society, 2021: 24(8): 1090 – 1100

[19] DHELIM S, KECHADI T, CHEN L, et al. Edge-enabled metaverse: The convergence of metaverse and mobile edge computing [EB/OL]. (2022-04-13) [2022-09-11]. https://arxiv.org/abs/2205.02764

[20] IBM. IBM blockchain service [EB/OL]. [2022-09-11]. https://www.ibm.com/blockchain

[21] AWS. AWS blockchain-as-a-service [EB/OL]. [2022-09-11]. https://aws.amazon.com/cn/blockchain

[22] ALIBABA. Alibaba blockchain solutions [EB/OL]. [2022-09-11]. https://cn.aliyun.com/solution/blockchain/tbes

[23] ORACLE. Oracle blockchain service [EB/OL]. [2022-09-11]. https://www.oracle.com/hk/blockchain/

[24] WOOD G. Ethereum: a secure decentralised generalised transaction ledger [J]. Computer science, 2014

[25] ANDROULAKI E, BARGER A, BORTNIKOV V, et al. Hyperledger fabric: a distributed operating system for permissioned blockchains [C]//Proceedings of the Thirteenth EuroSys Conference. ACM, 2018: 1 – 15. DOI: 10.1145/3190508.3190538

[26] AZURE M. Azure blockchain service [EB/OL]. [2022-09-11]. https://learn.microsoft.com/en-us/azure/confidential-ledger/

[27] WANG J, WU L B, CHOO K K R, et al. Blockchain-based anonymous authentication with key management for smart grid edge computing infrastructure [J]. IEEE transactions on industrial informatics, 2020, 16(3): 1984 – 1992. DOI: 10.1109/TII.2019.2936278

[28] ZHANG X D, LI R, CUI B. A security architecture of VANET based on blockchain and mobile edge computing [C]//Proceedings of 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN). IEEE, 2018: 258 – 259. DOI: 10.1109/HOTICN.2018.8605952

[29] CHENG G J, CHEN Y, DENG S G, et al. A blockchain-based mutual authentication scheme for collaborative edge computing [J]. IEEE transactions on computational social systems, 2022, 9(1): 146 – 158. DOI: 10.1109/TCSS.2021.3056540

[30] BROTSIS S, KOLOKOTRONIS N, LIMNIOTIS K, et al. On the security and privacy of hyperledger fabric: challenges and open issues [J]. IEEE world congress on services (SERVICES), 2020: 197 – 204

[31] DABHOLKAR A, SARASWAT V. Ripping the fabric: attacks and mitigations on hyperledger fabric [C]//International Conference on Applications and Techniques in Information Security. IEEE, 2019: 300 – 311

[32] DOCUMENTATION G. How to run a light node with geth [EB/OL]. [2022-09-11]. https://ethereum.org/en/developers/tutorials/run-light-node-geth/

[33] CASON D, FYNN E, MILOSEVIC N, et al. The design, architecture and performance of the tendermint blockchain network [C]//The 40th International Symposium on Reliable Distributed Systems (SRDS). IEEE, 2021: 23 – 33. DOI: 10.1109/SRDS53918.2021.00012

[34] CUI Z H, XUE F, ZHANG S Q, et al. A hybrid BlockChain-based identity authentication scheme for multi-WSN [J]. IEEE transactions on services computing, 2020, 13(2): 241 – 251. DOI: 10.1109/TSC.2020.2964537

[35] ZHU S, CAI Z, HU H, et al. Zkcrowd: a hybrid blockchain-based crowdsourcing platform [J]. IEEE transactions on industrial informatics, 2019: 16(6): 4196 – 4205

[36] TONG W, DONG X W, SHEN Y L, et al. CHChain: secure and parallel crowdsourcing driven by hybrid blockchain [J]. Future generation computer systems, 2022, 131: 279 – 291. DOI: 10.1016/j.future.2022.01.023

[37] NAKAMOTO S. Bitcoin: a peer-to-peer electronic cash system [EB/OL]. [2022-09-11]. https://nakamotoinstitute.org/bitcoin/

[38] CHAINSAFE. Ethereum javascript API [EB/OL]. [2022-09-11]. https://github.com/ChainSafe/web3.js

[39] REES K. Thousands of solana wallets drained in multimillion-dollar exploit [EB/OL]. [2022-09-16]. https://www.makeuseof.com/solana-wallets-drained-in-attack/

[40] LIU J Q, ZHAO Z H, CUI X, et al. A novel approach for detecting browser-based silent miner [C]//IEEE Third International Conference on Data Science in Cyberspace. IEEE, 2018: 490 – 497. DOI: 10.1109/DSC.2018.00079

[41] RAO V V, MARSHAL R, GOBINATH K. The IoT supply chain attack trends-vulnerabilities and preventive measures [C]//Proceedings of 2021 4th International Conference on Security and Privacy (ISEA-ISAP). IEEE, 2021: 1 – 4. DOI: 10.1109/ISEA-ISAP54304.2021.9689704

[42] FAROOQ M J, ZHU Q Y. IoT supply chain security: overview, challenges, and the road ahead [EB/OL]. [2022-09-16]. https://www.researchgate.net/publication/334658033_IoT_Supply_Chain_Security_Overview_Challenges_and_the_Road_Ahead

[43] ZAHAN N, ZIMMERMANN T, GODEFROID P, et al. What are weak links in the NPM supply chain? [C]//IEEE/ACM 44th International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP). IEEE, 2022: 331 – 340

[44] OPENSSF. Open source security metrics [EB/OL]. [2022-09-16]. https://metrics.openssf.org

[45] OSSF. Security scorecards-security health metrics for open source [EB/OL]. [2022-09-16]. https://hacker-gadgets.com/blog/2021/07/10/security-scorecards-security-health-metrics-for-open-source/

[46] SLOWMIST. A blockchain security firm established [EB/OL]. (2018-01-20) [2022-09-16]. https://www.slowmist.com/#services

[47] TANG X, ZHOU K, CHENG J, et al. The vulnerabilities in smart contracts: a survey [C]//International Conference on Artificial Intelligence and Security. ICAIS, 2021: 177 – 190

[48] JIANG B, LIU Y, CHAN W K. ContractFuzzer: fuzzing smart contracts for vulnerability detection [C]//Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering. ACM, 2018: 259 – 269. DOI: 10.1145/3238147.3238177

[49] CAO B, ZHANG Z H, FENG D Q, et al. Performance analysis and comparison of PoW, PoS and DAG based blockchains [J]. Digital communications and networks, 2020, 6(4): 480 – 485. DOI: 10.1016/j.dcan.2019.12.001

[50] CASTRO M, LISKOV B. Practical byzantine fault tolerance [EB/OL]. [2022-09-11]. https://www.geeksforgeeks.org/practical-byzantine-fault-tolerancepbft/

[51] YIN M F, MALKHI D, REITER M K, et al. HotStuff: BFT consensus with linearity and responsiveness [C]//Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing. ACM, 2019: 347 – 356. DOI: 0.1145/3293611.3331591

[52] MILLER A, XIA Y, CROMAN K, et al. The honey badger of BFT protocols [C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016: 31 – 42. DOI: 10.1145/2976749.2978399

[53] SANKAR L S, SINDHU M, SETHUMADHAVAN M. Survey of consensus protocols on blockchain applications [C]//The 4th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE, 2017: 1 – 5. DOI: 10.1109/ICACCS.2017.8014672

[54] NGUYEN G T, KIM K. A survey about consensus algorithms used in blockchain [J]. Journal of Information processing systems, 2018, 14(1): 101 – 128. DOI:10.3745/JIPS.01.0024

[55] FENG L, YANG Z X, GUO S Y, et al. Two-layered blockchain architecture for federated learning over the mobile edge network [J]. IEEE network, 2022, 36 (1): 45 – 51. DOI: 10.1109/MNET.011.2000339

[56] ASHERALIEVA A, NIYATO D. Reputation-based coalition formation for secure self-organized and scalable sharding in IoT blockchains with mobile-edge computing [J]. IEEE Internet of Things journal, 2020, 7(12): 11830 – 11850. DOI: 10.1109/JIOT.2020.3002969

[57] YUAN S J, LI J, LIANG J H, et al. Sharding for blockchain based mobile edge computing system: a deep reinforcement learning approach [C]//Proceedings of 2021 IEEE Global Communications Conference. IEEE, 2021: 1 – 6. DOI: 10.1109/GLOBECOM46510.2021.9685883

[58] HONG Z C, GUO S, LI P, et al. Pyramid: A layered sharding blockchain system [C]//IEEE Conference on Computer Communications. IEEE, 2021: 1 – 10. DOI: 10.1109/INFOCOM42981.2021.9488747

[59] WANG G, SHI Z J, NIXON M, et al. SoK: sharding on blockchain [C]//Proceedings of the 1st ACM Conference on Advances in Financial Technologies: ACM, 2019: 41 – 61. DOI: 10.1145/3318041.3355457

[60] SONNINO A, BANO S, AL-BASSAM M, et al. Replay attacks and defenses against cross-shard consensus in sharded distributed ledgers [C]//IEEE Euro-

pean Symposium on Security and Privacy (EuroS&P). IEEE, 2020: 294 – 308. DOI: 10.1109/EuroSP48549.2020.00026

[61] HAN R, YU J, LIN H, et al. On the security and performance of blockchain sharding [EB/OL]. [2022-09-11]. https://www. semanticscholar. org/paper/On-the-Security-and-Performance-of-Blockchain-Han-Yu/e7c5c3811973e26333f766012029d3069657871f

[62] KOKORIS-KOGIAS E, JOVANOVIC P, GASSER L, et al. OmniLedger: a secure, scale-out, decentralized ledger via sharding [C]//IEEE Symposium on Security and Privacy. IEEE, 2018: 583 – 598. DOI: 10.1109/SP.2018.000-5

[63] ZAMANI M, MOVAHEDI M, RAYKOVA M. RapidChain: scaling blockchain via full sharding [C]//ACM SIGSAC Conference on Computer and Communications Security. CCS, 2018: 931 – 948. DOI: 10.1145/3243734.3243853

[64] GUPTA S, SADOGHI M. Efficient and non-blocking agreement protocols [J]. Distributed and parallel databases, 2020, 38(2): 287 – 333. https://doi. org/10.1007/s10619-019-07267-w

[65] ABRAHAM I, CHAN T H H, DOLEV D, et al. Communication complexity of byzantine agreement, revisited [EB/OL]. [2022-09-11]. https://arxiv. org/abs/1805.03391

[66] GUPTA Y, SHOREY R, KULKARNI D, et al. The applicability of blockchain in the Internet of Things [C]//The 10th International Conference on Communication Systems & Networks (COMSNETS). IEEE, 2018: 561 – 564. DOI: 10.1109/COMSNETS.2018.8328273

[67] SHRESTHA R, BAJRACHARYA R, NAM S Y. Blockchain-based message dissemination in VANET [C]//IEEE 3rd International Conference on Computing, Communication and Security. IEEE, 2018: 161 – 166. DOI: 10.1109/CCCS.2018.8586828

[68] ZHANG P Y, PANG X, KUMAR N, et al. A reliable data-transmission mechanism using blockchain in edge computing scenarios [J]. IEEE Internet of Things journal, 2022, 9(16): 14228 – 14236. DOI: 10.1109/JIOT.2020.3021457

[69] SAAD M, THAI M T, MOHAISEN A. POSTER: deterring DDoS attacks on blockchain-based cryptocurrencies through mempool optimization [C]//Proceedings of the 2018 on Asia Conference on Computer and Communications Security. IEEE, 2018: 809 – 811. DOI: 10.1145/3196494.3201584

[70] LUO S C, SANG Y P, SONG M Y, et al. Preventing DDoS attacks on bitcoin memory pool by the dynamic fee threshold mechanism [C]//Parallel and distributed computing, applications and technologies, 2021: 172 – 184. DOI: 10.1007/978-3-030-69244-5_15

[71] SAAD M, NJILLA L, KAMHOUA C, et al. Mempool optimization for defending against DDoS attacks in PoW-based blockchain systems [C]//IEEE International Conference on Blockchain and Cryptocurrency. IEEE, 2019: 285 – 292. DOI: 10.1109/BLOC.2019.8751476

[72] SAAD M, KIM J, NYANG D, et al. Contra: mechanisms for countering Spam attacks on blockchain's memory pools [J]. Journal of network and computer applications, 2021, 179: 102971. DOI: 10.1016/j.jnca.2020.102971

[73] GUO Y H, ZHAO R, LAI S W, et al. Distributed machine learning for multiuser mobile edge computing systems [J]. IEEE journal of selected topics in signal processing, 2022, 16(3): 460 – 473. DOI: 10.1109/JSTSP.2022.3140660

[74] CHEN J G, LI K L, DENG Q Y, et al. Distributed deep learning model for intelligent video surveillance systems with edge computing [J]. IEEE transactions on industrial informatics, 2019, 99: 1. DOI: 10.1109/TII.2019.2909473

[75] SCHMOLL R S, PANDI S, BRAUN P J, et al. Demonstration of VR/AR offloading to mobile edge cloud for low latency 5G gaming application [C]//The 15th IEEE Annual Consumer Communications & Networking Conference. IEEE, 2018: 1 – 3. DOI: 10.1109/CCNC.2018.8319323

[76] ZUO Y, JIN S, ZHANG S, et al. Blockchain storage and computation offloading for cooperative mobile-edge computing [J]. IEEE Internet of Things Journal, 2021, 8(11): 9084 – 9098

[77] NGUYEN D C, PATHIRANA P N, DING M, et al. Secure computation offloading in blockchain based IoT networks with deep reinforcement learning [J]. Transactions on network science and engineering, 2021, 8(4): 3192 – 3208

[78] ZHAO C, ZHAO S N, ZHAO M H, et al. Secure Multi-Party Computation: Theory, practice and applications [J]. Information sciences, 2019, 476: 357 – 372. DOI: 10.1016/j.ins.2018.10.024

[79] ZHONG H, SANG Y, ZHANG Y, et al. Secure multi-party computation on blockchain: an overview [C]//International Symposium on Parallel Architectures, Algorithms and Programming. Springer, 2019: 452 – 460. DOI: 10.1007/978-981-15-2767-8_40

[80] ASWATHY S U, TYAGI A K, KUMARI S. The future of edge computing with blockchain technology: Possibility of threats, opportunities, and challenges [M]//Recent trends in blockchain for information systems security and privacy. Boca Raton: CRC Press, 2021: 261 – 292. DOI: 10.1201/9781003139737-18

[81] LU Y S, ZHANG J N, QI Y, et al. Accelerating at the edge: a storage-elastic blockchain for latency-sensitive vehicular edge computing [J]. IEEE transactions on intelligent transportation systems, 2022, 23(8): 11862 – 11876. DOI: 10.1109/TITS.2021.3108052

[82] CHATZIGIANNIS P, BALDIMTSI F, CHALKIAS K. Sok: blockchain light clients [C]//International Conference on Financial Cryptography and Data Security Cryptology. Springer, 2022: 615 – 641. DOI: 10.1007/978-3-031-18283-9_31

[83] MetaMask. The crypto wallet for Defi, Web3 Dapps and NFTs [EB/OL]. [2022-09-11]. https://metamask.io/

[84] ESKANDARI S, BARRERA D, STOBERT E, et al. A first look at the usability of bitcoin key management [C]//Proceedings 2015 Workshop on Usable Security. Internet Society, 2015: 55 – 63. DOI: 10.14722/usec.2015.23015

[85] HENDRIX C, LEWIS R. Survey on blockchain privacy challenges [EB/OL]. [2022-09-11]. http://ceur-ws.org/Vol-3031/paper_5.pdf

[86] BLACKSHEAR S, CHALKIAS K, CHATZIGIANNIS P, et al. Reactive key-loss protection in blockchains [C]//International Conference on Financial Cryptography and Data Security. Springer, 2021: 431 – 450. DOI: 10.1007/978-3-662-63958-0_34

[87] O'CONNOR R, PIEKARSKA M. Enhancing bitcoin transactions with covenants [C]//International Conference on Financial Cryptography and Data Security. Springer, 2017: 191 – 198

[88] MÖSER M, EYAL I, GÜN SIRER E. Bitcoin covenants [C]//International Conference on Financial Cryptography and Data Security. Springer, 2016: 126 – 141

[89] BARTOLETTI M, LANDE S, ZUNINO R. Bitcoin covenants unchained [EB/OL]. (2020-06-06)[2022-09-11]. https://arxiv.org/abs/2006.03918

[90] ZHOU T Q, SHEN J, REN Y J, et al. Threshold key management scheme for blockchain-based intelligent transportation systems [J]. Security and communication networks, 2021: 1864514. DOI: 10.1155/2021/1864514

[91] MA M X, SHI G Z, LI F H. Privacy-oriented blockchain-based distributed key management architecture for hierarchical access control in the IoT scenario [J]. IEEE access, 7: 34045 – 34059. DOI: 10.1109/ACCESS.2019.2904042

[92] ZHANG S J, LEE J H. A group signature and authentication scheme for blockchain-based mobile-edge computing [J]. IEEE Internet of Things journal, 2020, 7(5): 4557 – 4565. DOI: 10.1109/JIOT.2019.2960027

[93] PAAVOLAINEN S, CARR C. Security properties of light clients on the ethereum blockchain [J]. IEEE access, 8: 124339 – 124358. DOI: 10.1109/ACCESS.2020.3006113

[94] LETZ D. Blockquick: super-light client protocol for blockchain validation on constrained devices [EB/OL]. [2022-09-11]. https://eprint.iacr.org/2019/579

[95] YUAN L, TANG Q, WANG G L. Generic superlight client for permissionless blockchains [EB/OL]. [2022-09-11]. https://arxiv.org/abs/2003.06552

[96] JIANG S, CAO J N, ZHU J C, et al. PolyChain: a generic blockchain as a service platform [J]. Blockchain and trustworthy systems, 2021: 459 – 472. DOI: 10.1007/978-981-16-7993-3_36

[97] AMIRI M J, WU C, AGRAWAL D, et al. The bedrock of BFT: a unified platform for BFT protocol design and implementation [EB/OL]. (2022-05-09)[2022-09-11]. https://arxiv.org/abs/2205.04534v1

[98] CELESTIA. The first modular blockchain network [EB/OL]. [2022-09-11]. https://celestia.org/?ref=cypherhunter

[99] WERNER S M, PEREZ D, GUDGEON L, et al. SoK: decentralized finance (defi) [EB/OL]. (2021-01-21)[2022-09-11]. https://arxiv.org/abs/2101.08778v3

[100] SETHAPUT V, INNET S. Blockchain application for central bank digital currencies (CBDC) [C]//The Third International Conference on Blockchain Computing and Applications (BCCA). IEEE, 2021: 3 – 10. DOI: 10.1109/BCCA53669.2021.9657012

[101] KELKAR M, ZHANG F, GOLDFEDER S, et al. Order-fairness for byzantine consensus [C]//Annual International Cryptology Conference. CRYPTO, 2020: 451 – 480. DOI: 10.1007/978-3-030-56877-1_16

[102] REJIBA Z, MASIP-BRUIN X, MARÍN-TORDERA E. A survey on mobility-induced service migration in the fog, edge, and related computing paradigms [J]. ACM computing surveys, 2020, 52(5): 1 – 33. DOI: 10.1145/3326540

[103] BANDARA H D, XU X W, WEBER I. Patterns for blockchain data migration [C]//Proceedings of the European Conference on Pattern Languages of Programs 2020. ACM, 2020: 1 – 19. DOI: 10.1145/3424771.3424796

Biographies

CAO Yinfeng (csyfcao@comp.polyu.edu.hk) received his BS degree in information security from Xidian University (cyberspace security experimental class), China. He is currently a PhD candidate with the Department of Computing, The Hong Kong Polytechnic University, China. His research interests include blockchain systems and cryptography. He received the best paper reward from BlockSys in 2021.

CAO Jiannong is currently the Otto Poon Charitable Foundation Professor in data science and the Chair Professor of distributed and mobile computing in the Department of Computing, The Hong Kong Polytechnic University (PolyU), China. He is also the Dean of Graduate School, the director of Research Institute for Artificial Intelligence of Things (RIAIoT) in PolyU, and the director of the Internet and Mobile Computing Lab (IMCL) . He was the founding director and now the associate director of University's Research Facility in Big Data Analytics (UBDA) in PolyU. He served as the department head from 2011 to 2017. Prof. CAO is a member of Academia Europaea, a fellow of IEEE, a fellow of China Computer Federation (CCF), and an ACM distinguished member. His research interests include distributed systems and blockchain, wireless sensing and networking, big data and machine learning, and mobile cloud and edge computing.

WANG Yuqin is currently a PhD student at the Department of Computing, The Hong Kong Polytechnic University, China. Before that, he received a BE degree from Beijing Forestry University, China in 2021. His research interests include blockchain, edge computing, and wireless algorithm designs.

WANG Kaile received her BE degree in data science and big data technology from the University of International Business and Economics, China in 2021. She is currently a Mphil student at the Department of Computing, Hong Kong Polytechnic University, China. Her research interests include federated learning, data mining, and deep learning.

LIU Xun received her BS degree in computer science from Jilin University, China. She received her MS degree in computer science from the Institute of Information Engineering, Chinese Academy of Sciences. She is currently a PhD student in the Department of Computing, The Hong Kong Polytechnic University, China. Her research interests include cryptography and zero-knowledge proof.