A Survey on the Application of SGX in Blockchain Area

Hong Lei[1], Qinghao Wang[1,2], Wenbo Shi[2], and Zijian Bao[1(✉)]

[1] Oxford-Hainan Blockchain Research Institute, Chengmai 571924, China
{leihong,qinghao,zijian}@oxhainan.org
[2] Department of Computer Science and Engineering, Northeastern University, Shenyang 110001, China
shiwb@neuq.edu.cn
https://www.oxhainan.org

Abstract. As an emerging technology, blockchain is widely used in encrypted digital currencies and has an important impact in various fields such as finance, cloud storage, and Internet of things (IoT), etc. However, it faces various challenges in the process of its development and application: waste of resources, limited privacy protection, poor scalability, etc. Intel Software Guard Extensions (SGX), as a new trusted computing technology, brings solutions to the above challenges in the blockchain field. Based on the hierarchical structure of the blockchain, we are the first to systematically discuss the application status of SGX in the blockchian, including consensus layer, the ledger topology layer, the contract layer, and the application layer. Meanwhile, we summarize the advantages and challenges of SGX in the field of the blockchain, and look forward to the future development direction and the possible research topics.

Keywords: Blockchain · Intel SGX · Consensus algorithm · Smart contract · Privacy protection

1 Introduction

Blockchain is a technical solution that relies on distributed nodes to exchange, verify and store the network data without third parties. Bitcoin [1] is the first application of the blockchain, and its market capitalization is reaching 176 billion in June 2020 [2]. Ethereum [3] is another application of blockchain technology, which provides a platform for the operation of smart contracts to implement Turing-complete programming capabilities. At present, blockchain technology shows a wide range of application prospects. However, some issues restrict the development of the blockchain in nowadays. As we know, a large amount of power is consumed based on the proof of work mechanism. It restricts blockchain's application range severely. Meanwhile, the simple "Nakamoto" mechanism in blockchain can not completely guarantee the user's privacy. Some

Z. Zheng et al. (Eds.): BlockSys 2020, CCIS 1267, pp. 633–647, 2020.
https://doi.org/10.1007/978-981-15-9213-3_49

researches prove that the association between address and the user's identity can be obtained by analyzing the transactions on the blockchain [4,5]. Moreover, blockchain also suffers from other obstacles, such as poor scalability, lack of access to outside information, and centralization trend [6]. Therefore, how to solve the above issues becomes an essential step in the development of the blockchain.

Recently, employing Intel software guard extensions (SGX) to solve the problems in the blockchain has become a new research idea. SGX [7,8] is a trusted hardware technology developed by Intel corporation and has been added to Intel's CPU architecture. SGX provides a trusted memory range that preventing the code and data from being externally tampered and stolen. Meanwhile, the SGX built-in functions such as attestation mechanism, random number generation and monotonic counter provide the powerful efforts for solving security and privacy issues. The trusted execution environment (TEE) provided by SGX can ensure the correctness of execution (e.g., transaction verification, contract execution) and can protect private data from the outside world. An SGX-based party can be considered as a trusted party, which can replace complex cryptographic protocols to protect the security of schemes. Moreover, SGX can be used to simplify the process of protocols and enhance the security. However, due to the limitations of its design, SGX has certain deficiencies, such as performance load [9], memory restriction [10], side channel attacks [11]. These shortcomings limit the application of SGX technology in some scenarios and require researchers to combine effective scenarios to design effective solutions.

In general, TEE, especially Intel SGX technology, is a hot and highly developing field. Employing SGX to solve efficiency, privacy, and scalability problems in the blockchain still exist great challenges. In this work, we introduce existing schemes based on SGX in blockchain, abstract the functions and hidden dangers provided by SGX in these works, and suggest future directions in this area. In more details, our contributions are as follows:

1. The first summary of SGX-based research of blockchain. To the best of our knowledge, we propose the first systematic analysis on the application of SGX in blockchain system, which provides insight for employing SGX technology to solve the problems of the blockchain area. To introduce the existing works better, we divide the blockchain into six layers, which will be introduced in Sect. 2.1. We analyze the application of SGX in four of the above layers, which are associated with SGX.

2. Systematic analysis of the functions and challenges of SGX in blockchain. We summarize the practical functions of SGX employed in blockchain system. Meanwhile, we analyze the challenges caused by applying SGX to the blockchain and suggest the future directions in this area.

The rest of the paper is structured as follows. Section 2 presents an overview of blockchain's layers and Intel SGX technologies. Section 3 discusses the application of SGX in the consensus, ledger topology, contract, and application layers. We summary the advantages and disadvantages of SGX in the blockchain field in Sect. 4. Then, we discuss the open issues and show future directions in Sect. 5. Finally, we give the conclusion in Sect. 6.

2 Background

2.1 Blockchain Technology

Blockchain is a distributed ledger system. In the blockchain, nodes controlled by different users around the world form a vast P2P network to maintain the database system. The consistency is guaranteed by the consensus algorithm. For introducing the SGX-based schemes in blockchain field clearly, we introduce a hierarchical architecture of blockchain, as shown in Fig. 1.

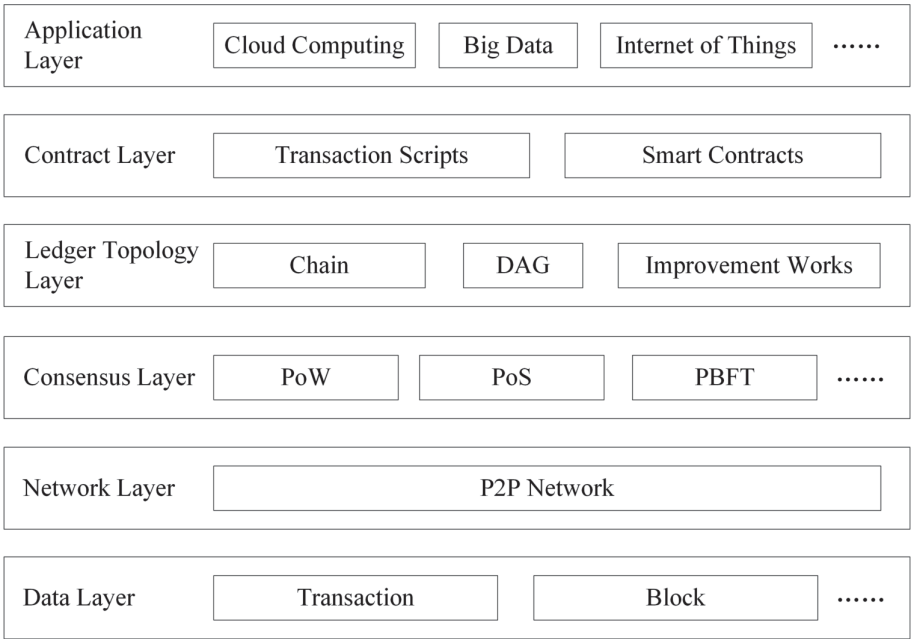

Fig. 1. The architecture of blockchain layer.

The Data Layer. The data layer mainly involves different data types recorded in blockchain, such as transactions, the hash value of block, smart contracts, etc.

The Network Layer. Nodes in the blockchain constitute a vast P2P network, and the functions of the nodes in the system are logically completely equal. Information is transmitted between the nodes in the form of flooding broadcasts.

The Consensus Layer. Blockchain uses a consensus algorithm to determine which node produces the next block, and maintaines the consistency of the blockchain. The mainstream consensus algorithms include Proof of Work (PoW) [1], Proof of Stake (PoS) [12], and Delegated Proof of Stake (DPoS) [13], etc. However, there are many limitations in existing consensus algorithm, such as resource waste (PoW), waste of resources (PoW), failure to provide provable security (PoS), and centralization trend (DPoS). We will discuss the SGX-based consensus algorithm in Sect. 3.1.

The Ledger Topology Layer. The ledger topology represents the state and structure of the data generated by the consensus layer, including chains, directed acyclic graphs (DAG), etc. Besides, the existing blockchain improvement efforts also produce some new ledger topologies by changing the organization, distribution, and status of the transactions, including payment channel [14], mix protocol [15], cross-chain transaction [16], etc. We will introduce the blockchain improvement schemes based on SGX in Sect. 3.2.

The Contract Layer. The contract layer includes simple trading scripts and smart contracts. In Bitcoin, the scripts contained in the transaction can preset the conditions for completion of the transaction. It provides simple programming features for the Bitcoin system (e.g., multi-signature transaction [17]). Compared to this simple program, smart contract implements Turing-complete scripting language, which can not only build complex trading logic but also implement any complex application. However, there are privacy issues in the smart contract. We will introduce the SGX-based solutions in Sect. 3.3.

The Application Layer. Blockchain provides a new solution to the problems in lots of fields, e.g., finance [41,42], cloud computing [43–45], IoT [46–48], etc. However, its limitations restrict its scope of application, such as most blockchain-based solutions inherit the inefficiency of blockchain. Combining blockchain with SGX to design better solutions may be an new direction. We will discuss it in Sect. 3.4.

2.2 Intel SGX

Intel Software Guard Extensions (SGX) technology is the extension of the Intel architecture to provide a trusted execution environment for applications. The computer system reserves a secure memory range for SGX. With a new set of instructions, developers can create secure containers for application in the secure memory. The integrity and confidentiality of the data in containers will be protected. SGX provides three major mechanisms: *secure container*, *attestation*, and *data sealing* [7,8].

Secure Container. SGX provides a secure container called ***enclave***, which protects the execution of code and data in it from external influences (including privileged software such as OS and Hypervisor). To protect enclave, the access of enclave will perform additional check by the hardware. The access instruction that fails the above verification will return a reference error that the memory address does not exist. Additional memory access check enables the enclave to be isolated from the outside world.

Attestation. Attestation mechanism can prove that an enclave has deployed the correct code and the SGX-based platform creating this enclave is credible. SGX proposes two attestation schemes: ***intra platform attestation*** and ***remote attestation***. Intra platform attestation is used by an enclave to prove to another enclave on the same platform, which can be used by calling the CPU instruction. Remote attestation process is based on intra platform attestation. It can be used to prove to the remote parties that it is trustworthy.

Data Sealing. Enclave data can be sealed out of secure memory for future use. SGX provides a special key called sealing key for data sealing to ensure the confidentiality and integrity of the data.

3 SGX-Based Schemes in the Blockchain Layers

3.1 SGX in the Consensus Layer

The principle and existing problems of these consensus algorithms have been introduced in Sect. 2.1. This section will discuss the schemes employing SGX to improve the existing consensus algorithms based on SGX.

Proof of Useful Work. PoW suffers from an enormous waste of resources, e.g., computing power, because nodes need to try to solve the computational puzzle each consensus round. To solve the problem, Zhang et al. [18] propose a new consensus scheme called Proof of Useful Work (PoUW). PoUW is an improvement to the PoW, which exploits SGX to transform the computing resources needed in PoW into useful works. In PoUW, every miner needs to support SGX, and get useful tasks from users each round. Miners will load those tasks into the enclave for execution. The enclave will determine if the execution wins the right to generate a new block in this round and provides certification. Finally, the winner can publish the new block and certification to the blockchain network by an agent. However, the security of PoUW will be destroyed by breaking the enclave in a miner node, because PoUW trusts the proof generated by each SGX-based miner node. For this problem, PoUW design a statistical analysis scheme based on newly generated blocks to detect whether SGX-based nodes are corrupted.

Secure Proof of Stake. PoS is another widely used consensus algorithm. In PoS, the possibility that a node obtains the right is related to the number and duration of assets it holds. Compared to PoW, PoS has less waste of resources and higher efficiency, but lower security [49,50]. Li et al. [19] propose the Secure Proof of Stake to improve the security of PoS. The scheme has the same way of obtaining rights generating new blocks as PoS, but it uses the nodes based on TEE (e.g., SGX) to provide security for the scheme. In the scheme, each node needs to generate a signature key pair in an enclave and configure the information in the blockchain network for joining the blockchain network. Since the signature key for the block is managed by enclave, the reliability of the block confirmation is ensured. Moreover, user's accounts are holden by the enclave, which improves security further. Considering the security problems of the PoS, the schemes implement a secure monotonic counter using SGX to ensure that the verifier generates at most one block at the existing block height. It prevents the verifier generating block for different branches to get more rewards at the same time.

Proof of Elapsed Time. Proof of Elapsed Time (PoET) [20] is a new consensus algorithm based on SGX, which is proposed in the Hyperledger Sawtooth Lake project. In the process of PoET, each node generates a random number, which represents the time that the node needs to wait. The node with the shortest waiting time will obtain the right to generate the block and receive the reward. The nodes of PoET only perform simple random number generation algorithm and corresponding waiting. Therefore, it does not need high computing resources. For security, the nodes need to execute protocol algorithm in the local enclave to ensure the correct generation of the random number and the correct execution of waiting time. The nodes will be verified with the remote attestation mechanism to ensure the reliability of the platform.

Proof of Luck. Based on the idea of PoET, Mitar et al. [21] propose a new consensus algorithm named Proof of Luck. The nodes in Proof of Luck are also based on TEE (e.g., SGX) and run the random number generation algorithm in the enclave to elect a leader. The smaller random number generated by the node, the sooner the new block generated by itself can be broadcast. After receiving the new block, nodes verify it and select the block with the smallest random number as the new block. To optimize performance, if nodes receive a block with a smaller random number before the broadcasting block of themselves, they will give up broadcasting their block to reduce network load. The scheme implements the SGX-based monotonic counter to prohibit concurrent calls of the enclave, which prevents individual nodes from running multiple consensus nodes concurrently to increase their competitiveness.

3.2 SGX in the Ledger Topology Layer

This section introduces the schemes based on SGX to enhance the improvement works of the blockchain.

Payment Channel. Payment channel [14] is one of the solutions to improve the throughput of blockchain. It can implement multiple off-chain transactions and only synchronize a settlement transaction in the blockchain. However, existing payment channel schemes rely on complex cryptographic protocols to ensure security, which results in inefficiencies and unfriendly to users. Lind et al. [22] propose Teechain, a secure payment channel solution. In Teechain, both parties that build payment channels are based on SGX. The enclave of both parties maintain the balance information, the address of transactions, and the corresponding private key. Before building the payment channel, both parties must deposit the funds that determining the transaction capacity of the payment channel. With remote attestation, each party constructs a secure communication channel for synchronizing the balance. During the transaction, the payment message is encrypted by the enclave and sent to the other party. The other party will receive the payment message and updates the balance. The balance maintained by the enclave of both parties and will not be affected by the users. Since

both enclaves hold the key for the settlement transaction and the transaction based on the hash time lock, both users can settle the transaction separately after the timeout. Teechain takes the way that each user can only send the payment messages to ensure that neither user will reject the other party's messages to destroy the protocol. Meanwhile, Teechain employs hardware-based monotonic counters to prevent the replay attacks.

Mix Protocol. Transaction mix protocol is one of the solutions to solve the privacy problem in the blockchain. Most mix schemes use the centralized mix server model [15], which uses a single bitcoin address to receive the same amount of funds from multiple user addresses. Then, the server transfers the funds to the destination address. Since all user addresses have the same probability of being traded with the destination address, it is difficult for attackers to associate user and destination address. However, the centralized mixer has huge rights, and it is difficult for the protocol to constrain the malicious behavior of mixers. Tran et al. [23] propose a new privacy protection scheme. The scheme is based on the centralized model. The centralized mixer is based on SGX, and the mix operation will be load into the enclave to effectively prevent the malicious mix server from malicious behavior. To mix transactions, a user needs to construct a transaction (including information such as funds and destination address) to transfer funds to mixer's address. Enclave periodically get the transactions related to the mix and constructs the corresponding transfer transactions. Finally, the mixer will broadcast the transaction to the blockchain network. Since the mixer's address is generated and saved in the enclave, it can be ensured that the funds can only be spent by the enclave. The establishment of the secure communication channel between the user and the mixer ensures that the privacy of the mix information.

Cross-Chain Transaction. Atomic Cross-Chain Swaps (ACCS) achieve untrusted cross-chain transactions [16]. However, it relies on the parties to interact with the transactions continuously to ensure the security of the protocol. It is not only unfriendly to the user, but the complex interaction process seriously affects the performance of the protocol. Tesseract [24] is a cross-chain trading scheme that employing the secure execution features of SGX to implement real-time cross-chain transactions. In Tesseract, there is a central exchange based on SGX. The user who needs the service registers a Tesseract account and deposits a certain amount of balance into the exchange's address. The exchange enclave records the private key of the exchange's address and the balance of all users. In the transaction, the user makes orders by offering requests to the exchange. After receiving the message, the Tesseract enclave issues an order that is anonymous and visible to everyone. Other users choose the appropriate order to trade, and enclave updates the user's account balance based on the transaction information. Tesseract enclave will periodically synchronize settlement transactions to the blockchain network. The user can utilize the remote attestation mechanism to verify the program's validity in the enclave and build a secure channel to ensure the privacy of their transactions.

Light-Weight Client. Some researchers [25,26] propose the blockchain light-weight client schemes to support the node based on resource-constrained devices (e.g., mobile phones). In the scheme, there are two types of nodes: light nodes and full nodes. The light nodes only need to save a small amount of data on the chain and outsource most of the verification and storage work to full nodes. However, when full nodes work for light nodes, it needs to know all the transaction information from light nodes, which seriously affects the privacy of users. Matetic et al. [27] propose an SGX-based light-weight client scheme to protect the privacy of users. In the scheme, full nodes are based on SGX and load the transaction verification process into the enclave. When a light node sends a request to a full node, the enclave on the full node will scan the blockchain data and reply the Merkle path of the block. The light node verifies the correctness of the transaction through the Merkle path and the block header. They [27] also provides a variant scheme to improve the efficiency of verification. In the variant scheme, the enclave on the full node maintains a special version of the Unspent Transaction Outputs (UTXO) database. When receiving the verification request from a light node, the enclave will access the database and return the corresponding result directly. In both schemes, light nodes and the enclaves can construct the secure communication channel to protect the privacy of user.

3.3 SGX in the Contract Layer

The contract layer includes trading script and smart contract. The trading script can only build simple trading logic, while the smart contract has more programming power than general trading scripts. However, the contract codes on the blockchain are visible to all nodes, which undoubtedly affect the privacy of the smart contract. Some researchers [4,5] prove the feasibility of de-anonymization attacks by analyzing the transaction structure of blockchain. This section introduces SGX-based schemes to solve the privacy issue in the smart contract.

Kosba et al. [28] propose a smart contract system to protect the privacy of smart contracts. They design a manager to execute part of the user's smart contract, which includes private information. The manager will construct a zero-knowledge proof to prove the execution results. To ensure that manager is trusted and does not reveal sensitive data, they recommend that the manager can be loaded in a TEE (e.g., SGX). Yuan et al. [29] propose a scheme that protecting the security and confidentiality of smart contracts without breaking the integrity of existing blockchains. The scheme establishes a TEE-distributed storage platform (TEE-DS) as the execution platform of the smart contract. TEE-DS consists of the worker nodes based on SGX. To ensure the confidentiality of the smart contract code and data, the user needs to establish a secure channel with the TEE-DS before transferring the contract. The smart contract code will be transmitted using the secure channel. Users can be free to choose whether to become a worker node, which guarantees the scalability of the system. Cheng et al. [30] propose Ekiden to protect the privacy of smart contracts. Ekiden also separates execution of smart contact from consensus operations. The execution of smart contract is responsible for compute nodes, and consensus operations are

responsible for consensus nodes. The consensus nodes are responsible for maintaining the blockchain system and updating the state of the smart contract. The compute nodes are based on SGX and will execute the smart contract in the enclave. Any node that supports SGX can join the system as a compute node. To reduce the impact of failed compute nodes, the scheme designed a key management protocol based on secret sharing [31,32]. Based on the design idea of Ekiden [30], Das et al. [33] implement a smart contract execution scheme based on Bitcoin. In the scheme, all users must submit the deposit into their contract before their contracts are executed, which is different from Ekiden [30]. At the same time, the execution node based on SGX must submit the margin equal to the sum of deposits. If the protocol fails, the party that misbehaves will lose the deposit. However, the scheme only supports the limited types of contracts and has security issues (e.g., multi-party collusion to secure margin).

3.4 SGX in the Application Layer

In this section, we will discuss the schemes in different application fields such as finance, cloud storage, and Internet of things (IoT) [34–36].

Distributed Cloud Computing Service. Most of the existing cloud computing models are based on centralized service models, which brings vast pressure of equipment to cloud computing servers and the trust problem between servers and users. Al-Bassam et al. [34] propose a distributed cloud computing solution, which allows execution nodes to rent their own trusted calculation time. It designs a fair trade protocol that combines the SGX remote attestation mechanism with smart contracts to ensure fairness in rental service. With the secure execution environment of the SGX and smart contract, the scheme ensures the correct execution of transactions and user' code. Meanwhile, it employs the distributed cloud computing model to ensure that users can continue to execute on other execution nodes if the current execution node is corrupted or offline. However, the scheme requires the separate construction of the payment channels between the user and the execution node, and it needs to maintain continuous communication during the rental process, which limits the availability of the scheme.

Data Ownership and Privacy Protection Data ownership and privacy protection have become one of the key researches in the era of big data. The existing schemes, including data access control [37] and data anonymization [38], cannot guarantee the proper use of data by authorized users. Yang et al. [35] combine smart contract with TEE (e.g., SGX) to enable privacy of data, which enables users to control other people's use of their private data. In the Privacy Guard, data owners use smart contracts to set usage policies for data, including data consumers, data usage conditions, and the purpose of the data. At the same time, the data usage record is stored on the blockchain to ensure the unchangeability

and traceability. Privacy Guard delivers smart contract to the off-chain execution engine based on TEE. The correctness of execution does not depend on the consensus algorithm and does not require all blockchain nodes to execute the contract, which improves system efficiency. In general, Privacy Guard leverages TEE's isolation feature to protect the execution of smart contracts while reducing the consumption of smart contract consensus algorithms. The TEE remote attestation mechanism solves the user's trust in data storage.

IoT Data Security. Blockchain provides a new solution to the secure storage and management of IoT devices. However, the huge data flow of IoT far exceeds the throughput of existing blockchain systems. Meanwhile, the hybrid storage architecture, which stores data digests on the chain and stores the original data out of the chain, cannot guarantee the integrity and privacy of the IoT data. Ayoade et al. [36] propose a decentralized IoT data management solution. The scheme employs smart contracts to achieve data access control and SGX technology to store IoT data securely. The IoT device in the scheme is registered in the blockchain through the IoT gateway. It uses the smart contract to set the data access control algorithm to ensure that only authorized users can access the data. Moreover, the scheme uses the hybrid storage architecture to store data digests in blockchains, and the original data is stored out of the chain. To protect the off-chain data, the data will be encrypted by SGX. When a user wants to get the original data, he first proves his authority to the blockchain. A certificate will be obtained from the blockchain and submitted to the storage platform. The enclave of the storage platform performs integrity verification on the certificate and returns data to the user.

4 Discussion

At present, SGX has many applications in the blockchain field. Through the analysis of the previous sections, the functions of SGX in the field of blockchain can be summarized as follows.

1. *Secure execution.* An enclave is isolated from the external environment, so the execution logic of the code loaded into the enclave cannot be tampered by the external environment, which ensures the correctness of internal execution.
2. *Privacy protection.* The data in the enclave cannot be accessed and tampered by the outside world, so the privacy of sensitive data generated during execution can be effectively protected. Besides, the secure channel built between the enclave and the other party also guarantees the privacy of the data passed into the enclave.
3. *Simplify the protocol process.* Most cryptographic protocols often employ complex cryptographic tools, and cumbersome protocol flows to ensure security. The SGX-based role can act as a trusted party in the protocol to reduce the use of complex cryptographic tools and to simplify the protocol process, thereby improving the efficiency of the schemes.

4. *Trusted functions based on SGX.* Some trusted functions provided by SGX (such as trusted monotonic counters, trusted random number generation, etc.) can be implemented to provide trusted components for the solution, thereby improving the security of the scheme.

Obviously, SGX provides practical help to the research field of blockchain. Of course, there are still some problems in the application of SGX in the blockchain. Some of the issues are caused by the unique scenes of the blockchain, and the others are caused by defects of SGX. The details are as follows.

1. *Controlled communication.* The communication of SGX is controlled by the platform owner, so it is easy for the platform owner to intercept input of an enclave. Of course, the information passed into an enclave can be encrypted to ensure data security, but the interception cannot be stopped. It may lead to some network communication-based attacks (such as denial of service attacks, Eclipse attacks, replay attacks). Therefore, researchers have to design reasonable protocol to decrease the possibility that the SGX platform owner will intercept or tamper the input of the enclave. A reference is the Teechain [14], which introduces the beneficial results for an SGX platform owner to receive the correct inputs.
2. *Single point attack.* Most of the SGX-based solutions rely on the credibility of SGX. Once the SGX-based role is compromised, it is easy to cause the entire solution to crash. The problem is very conspicuous in the consensus schemes, which employ the SGX-based roles as the nodes in a P2P network (e.g., PoET [20], PoLK [21]). In those schemes, the SGX-based roles may maintain the large value relationship, which stimulates the attacker to launch attacks on the SGX, even the physical means. To tolerate such attack, a solution is to decentralize the benefit to reduce the possibility of attack. Meanwhile, it is possible that multiple SGX-based nodes cooperate to mitigate the attacks.
3. *Side channel attacks.* In recent years, some side channel attacks on SGX are exploded, which indicates that SGX indeed have some security holes. Although these attacks are relatively difficult to exploit, the attacker may use these means to attack the SGX with sufficient profit. To solve the problem, we can consider combining the scheme with the side channel attack defense scheme (such as Oblivious Random Access Machine [39], Address Space Layout Randomization [40]) to improve the security.

5 Future Directions

SGX provides the new solution to the problems of efficiency, privacy, and scalability in the blockchain. It has attracted much attention to academia and industry. We believe the following aspects should be noted in future research.

Security in Consensus Algorithms. In the blockchain, the consensus mechanism, while considered as a way to ensure fairness and trust in an untrusted system, provides a target for would-be attackers. We discuss the schemes using SGX to eliminate the attacks in existing consensus algorithms in Sect. 3.1. However, SGX has the potential to solve the more attacks in the consensus layer. For example, selfish mining attack is an attack strategy in PoW, which permits the miner to obtain more rewards of creating blocks. To implement the attack, a miner is not broadcast the generated block immediately and continue to generate the next block in a round, until a new block is broadcasted. Obviously, the miner based on SGX can be forced to broadcast the generated block, which prevents against selfish mining attack. Thus, it is a potential direction to solve the more attacks in the existing consensus algorithm using SGX.

Privacy Protection and Supervision. Privacy protection technology may provide a safeguard for the unlawful act. For example, offenders can anonymously achieve money laundering by anonymous digital currency. Thus, a splendid blockchain system should supervise the criminal acts while provides privacy protection. In this scenario, the SGX-based role can act as a qualified supervisor to hold the secret that can track the users. Thus, it is also a potential direction to build a trusted supervisor based on SGX.

6 Conclusion

This paper analyzes the application of SGX in the field of blockchain. Firstly, the blockchain layers and SGX technology are introduced. Secondly, the problems of blockchain and SGX-based solutions in the consensus layer, the ledger topology layer, the contract layer, and the application layer are elaborated. Finally, the functions of SGX in the field of blockchain are summarized, and the future directions have prospected.

Acknowledgements. This study is supported by Oxford-Hainan Blockchain Research Institute, the National Science Foundation of China (No. 61472074, U1708262) and the Fundamental Research Funds for the Central Universities (No. N172304023).

References

1. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2008)
2. Bitcoin market value (2019). https://coinmarketcap.com/zh/currencies/bitcoin/
3. Buterin, V.: A next-generation smart contract and decentralized application platform. White paper (2014)

4. Bonneau, J., et al.: Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In: IEEE Symposium on Security and Privacy (2015)
5. Meiklejohn, S., et al.: A fistful of bitcoins: characterizing payments among men with no names. In: Proceedings of the 2013 Conference on Internet Measurement Conference. ACM (2013)
6. Zheng, Z., Xie, S., Dai, H., Wang, H., Chen, X.: Blockchain challenges and opportunities: a survey. Int. J. Web Grid Serv. (IJWGS) **14**(4), 352–375 (2018)
7. McKeen, F., et al.: Innovative instructions and software model for isolated execution. Hasp@isca10.1 (2013)
8. Anati, I., et al.: Innovative technology for CPU based attestation and sealing. In: Proceedings of the 2nd International Workshop on Hardware and Architectural Support for Security and Privacy, vol. 13 (2013)
9. Arnautov, S., et al.: SCONE: secure linux containers with intel SGX. In: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI) (2016)
10. Wang, W., et al.: Leaky cauldron on the dark land: understanding memory side-channel hazards in SGX. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. ACM (2017)
11. Hunt, T., et al.: Ryoan: a distributed sandbox for untrusted computation on secret data. ACM Trans. Comput. Syst. (TOCS) **35**(4), 13 (2018)
12. Seijas, P.L., Thompson, S.J., McAdams, D.: Scripting smart contracts for distributed ledger technology. IACR Cryptology ePrint Archive 2016:1156 (2016)
13. Seijas, P.L., Thompson, S.J., McAdams, D.: Scripting smart contracts for distributed ledger technology. IACR Cryptol. ePrint Archieve Technology (2016)
14. Poon, J., Dryja, T.: The bitcoin lightning network: Scalable off-chain instant payments (2016)
15. Valenta, L., Rowan, B.: Blindcoin: blinded, accountable mixes for bitcoin. In: Brenner, M., Christin, N., Johnson, B., Rohloff, K. (eds.) FC 2015. LNCS, vol. 8976, pp. 112–126. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-48051-9_9
16. Nolan, T.: Alt chains and atomic transfers. In: Bitcoin Forum, May 2013
17. Okupski, K.: Bitcoin developer reference. In: Eindhoven (2014)
18. Zhang, F., et al.: REM: resource-efficient mining for blockchains. In: 26th USENIX Security Symposium (USENIX Security 17) (2017)
19. Li, W., Andreina, S., Bohli, J.-M., Karame, G.: Securing proof-of-stake blockchain protocols. In: Garcia-Alfaro, J., Navarro-Arribas, G., Hartenstein, H., Herrera-Joancomartí, J. (eds.) ESORICS/DPM/CBT -2017. LNCS, vol. 10436, pp. 297–315. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67816-0_17
20. Intel Corporation. Sawtooth lake (2016)
21. Milutinovic, M., et al.: Proof of luck: an efficient blockchain consensus protocol. In: Proceedings of the 1st Workshop on System Software for Trusted Execution. ACM (2016)
22. Lind, J., et al. Teechain: a secure payment network with asynchronous blockchain access. In: Proceedings of the 27th ACM Symposium on Operating Systems Principles. ACM (2019)
23. Tran, M., et al.: Obscuro: a bitcoin mixer using trusted execution environments. In: Proceedings of the 34th Annual Computer Security Applications Conference. ACM (2018)
24. Tran, M., et al.: Obscuro: a bitcoin mixer using trusted execution environments. In: Proceedings of the 34th Annual Computer Security Applications Conference. ACM (2018)

25. Bünz, B., et al.: Flyclient: super-light clients for cryptocurrencies. IACR Cryptology ePrint Archive 2019:226 (2019)
26. BitcoinJ (2018). https://bitcoinj.github.io/
27. Matetic, S., et al.: BITE: bitcoin lightweight client privacy using trusted execution. In: 28th USENIX Security Symposium (USENIX Security 19) (2019)
28. Kosba, A., et al.: Hawk: the blockchain model of cryptography and privacy-preserving smart contracts. In: 2016 IEEE symposium on security and privacy (SP). IEEE (2016)
29. Yuan, R., et al.: Shadoweth: private smart contract on public blockchain. J. Comput. Sci. Technol. **33**(3), 542–556 (2018)
30. Yuan, R., et al.: Shadoweth: private smart contract on public blockchain. J. Comput. Sci. Technol. **33**(3), 542–556 (2018)
31. Herzberg, A., Jarecki, S., Krawczyk, H., Yung, M.: Proactive secret sharing or: how to cope with perpetual leakage. In: Coppersmith, D. (ed.) CRYPTO 1995. LNCS, vol. 963, pp. 339–352. Springer, Heidelberg (1995). https://doi.org/10.1007/3-540-44750-4_27
32. Schultz, D., Liskov, B., Liskov, M.: MPSS: mobile proactive secret sharing. ACM Trans. Inf. Syst. Secur. (TISSEC) **13**(4), 1–34 (2010)
33. Das, P., et al.: FastKitten: practical smart contracts on bitcoin. IACR Cryptology ePrint Archive 2019 (2019)
34. Al-Bassam, M., et al.: Airtnt: fair exchange payment for outsourced secure enclave computations. arXiv preprint arXiv:1805.06411 (2018)
35. Xiao, Y., et al.: Enforcing Private Data Usage Control with Blockchain and Attested Off-chain Contract Execution. arXiv preprint arXiv:1904.07275 (2019)
36. Ayoade, G., et al.: Decentralized IoT data management using blockchain and trusted execution environment. In: 2018 IEEE International Conference on Information Reuse and Integration (IRI). IEEE (2018)
37. Yu, S., et al. Achieving secure, scalable, and fine-grained data access control in cloud computing. In: 2010 Proceedings IEEE INFOCOM. IEEE (2010)
38. Li, N., Li, T., Venkatasubramanian, S.: t-closeness: privacy beyond k-anonymity and l-diversity. In: 2007 IEEE 23rd International Conference on Data Engineering. IEEE (2007)
39. Ahmad, A., et al.: OBFUSCURO: a commodity obfuscation engine on intel SGX. In: NDSS (2019)
40. Seo, J., et al.: SGX-shield: enabling address space layout randomization for SGX programs. In: NDSS (2017)
41. Tapscott, A., Tapscott, D.: How blockchain is changing finance. Harvard Bus. Rev. **1**(9), 2–5 (2017)
42. Hofmann, E., Strewe, U.M., Bosia, N.: Supply Chain Finance and Blockchain Technology: The Case of Reverse Securitisation. Springer, New York (2017)
43. Zheng, B.-K., et al.: Scalable and privacy-preserving data sharing based on blockchain. J. Comput. Sci. Technol. **33**(3), 557–567 (2018)
44. Gaetani, E., et al.: Blockchain-based database to ensure data integrity in cloud computing environments (2017)
45. Liang, X., et al.: Provchain: a blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability. In: Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. IEEE Press (2017)
46. Dorri, A., et al.: Blockchain for IoT security and privacy: the case study of a smart home. In: 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE (2017)

47. Zhang, Y., Wen, Jiangtao: The IoT electric business model: using blockchain technology for the internet of things. Peer-to-Peer Networking Appl. **10**(4), 983–994 (2016). https://doi.org/10.1007/s12083-016-0456-1
48. Sun, J., Yan, J., Zhang, K.Z.K.: Blockchain-based sharing services: what blockchain technology can contribute to smart cities. Financ. Innovation **2**(1), 26 (2016)
49. Martinez, J.: Understanding proof of stake: the nothing at stake theory (2019)
50. Gaži, P., Kiayias, A., Russell, A.: Stake-bleeding attacks on proof-of-stake blockchains. In: 2018 Crypto Valley Conference on Blockchain Technology (CVCBT). IEEE (2018)