Peng, Li; Feng, Wei; Yan, Zheng; Li, Yafeng; Zhou, Xiaokang; Shimizu, Shohei

Privacy Preservation in Permissionless Blockchain: A Survey

Published in:
Digital Communications and Networks

DOI:
10.1016/j.dcan.2020.05.008

Published: 01/08/2021

Document Version
Publisher's PDF, also known as Version of record

Digital Communications and Networks 7 (2021) 295–307

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

Privacy preservation in permissionless blockchain: A survey

Li Peng [a], Wei Feng [a], Zheng Yan [a,b,*], Yafeng Li [c], Xiaokang Zhou [d,e], Shohei Shimizu [d,e]

[a] School of Cyber Engineering, Xidian University, Xi'an, 710071, China
[b] Department of Communications and Networking, Aalto University, Espoo, 02150, Finland
[c] The 20th Research Institute of China Electronics Technology Group Corporation, Xi'an, 710061, China
[d] Faculty of Data Science, Shiga University, Hikone, Japan
[e] RIKEN Center for Advanced Intelligence Project, Tokyo, Japan

ARTICLE INFO

Keywords:
Blockchain
Privacy
Cryptocurrency
Anonymity

ABSTRACT

Permissionless blockchain, as a kind of distributed ledger, has gained considerable attention because of its openness, transparency, decentralization, and immutability. Currently, permissionless blockchain has shown a good application prospect in many fields, from the initial cryptocurrency to the Internet of Things (IoT) and Vehicular Ad-Hoc Networking (VANET), which is considered as the beginning of rewriting our digital infrastructure. However, blockchain confronts some privacy risks that hinder its practical applications. Though numerous surveys reviewed the privacy preservation in blockchain, they failed to reveal the latest advances, nor have they been able to conduct a unified standard comprehensive classification of the privacy protection of permissionless blockchain. Therefore, in this paper, we analyze the specific characteristics of permissionless blockchain, summarize the potential privacy threats, and investigate the unique privacy requirements of blockchain. Existing privacy preservation technologies are carefully surveyed and evaluated based on our proposed evaluation criteria. We finally figure out open research issues as well as future research directions from the perspective of privacy issues.

1. Introduction

As the core technology of cryptocurrencies and various decentralized applications, the blockchain has attracted considerable attention in both academia and industry. It is a distributed database or public transaction ledger shared by all participants. The security of blockchain relies on the underlying data encryption, time stamping, distributed consensus, and incentive mechanism, rather than a Trusted Third Party (TTP) [1]. It can solve the problem of trust establishment between nodes in the decentralized system through verification and consensus mechanism, and thereby distrusted users can complete transactions or data exchange without a trusted third party. The emergence of Ethereum enables users to run smart contracts on the blockchain, thereby significantly expanding the scope of blockchain applications. Currently, researchers have applied blockchain in various systems, such as the Internet of Things (IoT) [2–4], Fog Computing [5], Vehicular Ad-Hoc Network (VANET) [6,7], and smart city [8,9], etc. [10–13]. In summary, blockchain has shown a promising prospect during the past years.

Blockchain can be roughly divided into permissioned blockchain and permissionless blockchain. Among them, permissioned blockchain only allows authorized entities to work as consensus nodes and access data in the blockchain. Differently, permissionless blockchain allows every entity to join and leave freely [14]. Besides, data in the permissionless blockchain is transparent to all entities for public verification. Compared with permissioned blockchain, permissionless blockchain faces more risky privacy issues since in the permissioned blockchain, it is easier to ensure privacy with access control. For the permissionless blockchain, although the openness and transparency of the permissionless blockchain help improve its trust, the disclosure of transaction content may lead to crucial privacy leaks, especially when it is applied in scenarios like Mobile CrowdSourcing (MCS) and the Internet of Things (IoT), where transactions may contain sensitive information of users. Apart from direct privacy leakage, attackers can track the transactions of a user through its address, analyze the transaction rules, obtain the association between the user transaction addresses, and infer its true identity with external information of the network [15]. Even worse, the transparency of permissionless blockchain may result in the misuse of user data. For example, competitive enterprises or individuals can benefit from

* Corresponding author.
E-mail addresses: pengli_email@163.com (L. Peng), weifeng.ft@foxmail.com (W. Feng), zyan@xidian.edu.cn (Z. Yan), xxddxdd@yeah.net (Y. Li), zhou@biwako.shiga-u.ac.jp (X. Zhou), shohei-shimizu@biwako.shiga-u.ac.jp (S. Shimizu).

https://doi.org/10.1016/j.dcan.2020.05.008
Received 4 December 2019; Received in revised form 26 May 2020; Accepted 27 May 2020
Available online 25 June 2020
2352-8648/

analyzing the transaction data or obtain sensitive information of users like user habit. Therefore, permissionless blockchain confronts significant privacy risks, which dramatically limits its practical application.

However, privacy preservation in the permissionless blockchain is not trivial [16]. Different from centralized systems, permissionless blockchain is an open and decentralized system that lacks a powerful authority for system maintenance and privacy insurance. As a result, traditional privacy solutions are not applicable in the blockchain. Besides, the openness of the permissionless blockchain makes it easier for an attacker to intrude into the system and compromise a number of nodes. Additionally, most of the existing permissionless blockchain systems suffer from low efficiency, high communication overhead, low throughput, and high confirmation latency [17]. Even the latest consensus mechanisms, e.g., Algorand [18] Bitcoin-ng [19], significantly improve the performance of permissionless blockchain, the throughput still cannot support computationally expensive cryptographic operations for privacy preservation. Therefore, it is challenging to achieve practical privacy preservation in permissionless blockchain systems.

To assist future works in the privacy preservation of the permissionless blockchain, we survey the privacy solutions published in high-level journals and conferences to trigger open issues and significant future research directions. For easy presentation, in our paper, we refer to the blockchain as the permissionless blockchain. There have been some investigations into privacy issues in blockchain [20–24]. Feng et al. [20] provided a discussion on various privacy-preservation methods employed in blockchain along with preliminary knowledge of the technical background of these techniques and proposed a list of future research directions. Yang et al. [21] gave a comprehensive technical survey and discussed the efficiency of various methods, which is a bit outdated. Conti et al. [22] reviewed the security and privacy aspects of Bitcoin-like systems and discussed various threats to user security and transaction anonymity, which restricted the applicability of cryptocurrencies in real-world applications and services. In Ref. [23], Zhang et al. provided a technical survey of blockchain security-enhancing technology and insinuated some open challenges. Li et al. [24] systematically overviewed and analyzed the security challenges of blockchain. They also described and evaluated existing solutions that addressed some existing research problems and gave a list of open issues.

However, the development of blockchain itself and technologies for privacy preservation in the blockchain is quite rapid, which makes these surveys cannot well reveal the latest research status or fail to review works with a comprehensive classification. For the emerging blockchain-based scalable payment methods, off-chain payment channels, and blockchain-based computation platforms, smart contracts, there still lacks a systematic survey to thoroughly discuss their privacy challenges and solutions. The concrete comparison of these surveys is demonstrated in Table 1. In summary, there still lacks a systematic survey on the latest advance of privacy preservation in the blockchain.

Different from the above studies, this paper makes a comprehensive investigation and comparison of privacy preservation schemes in the blockchain based on a number of privacy requirements. Considering the fact that all the information in blockchain is delivered and recorded through transactions, and various decentralized applications are built upon the smart contract as the trusted computation platform, this work will focus on transaction privacy and smart contract privacy, which are two main privacy issues in blockchain systems towards practical applications. Problems beyond these two aspects (such as privacy in the consensus process) are out of the scope of our discussion. We analyze the issues in blockchain according to its architecture, specific characteristics, and potential threats. We propose a series of evaluation criteria from the view of both privacy preservation and availability, which enable us to analyze the existing works systematically. Furthermore, we propose future research directions. Specifically, contributions of this paper can be summarized as below:

1) We summarize the system model and application scenarios of permissionless blockchain and analyze its unique characteristics.
2) Based on the characteristics of the blockchain, we analyze the privacy issues and then summarize the potential threats to privacy in the blockchain. A series of requirements for privacy preservation in the blockchain is proposed to evaluate existing privacy solutions.
3) We employ the proposed requirements as criteria for evaluating and comparing privacy countermeasures published in influential journals and conferences. We summarize the advantages and disadvantages of each work, based on which we propose unsolved open research issues, a series of future research directions, and provide instruction on future research of privacy preservation in the blockchain.

The rest of this paper is organized as follows. Section 2 outlines the basic architecture of the blockchain and summarizes its unique characteristics. Section 3 provides a detailed analysis of the privacy threats and privacy protection requirements in the blockchain. In Section 4, we classify the privacy preservation schemes into different categories according to their design goals and technologies and make a comprehensive analysis of the proposed requirements as evaluation criteria. Section 5 gives our summarization and the discussion of future research direction. Finally, Section 7 concludes this paper.

2. Overview of blockchain technologies

In this section, we present a basic introduction to blockchain, including its definition, development, application scenarios, and system model, and analyze its unique characteristics.

2.1. Introduction to blockchain

This section presents the system model and the unique characteristics of permissionless blockchain.

2.1.1. System model

There are two types of nodes in a permissionless blockchain, i.e., miners and users, and every node can choose to be a miner or a user freely. The miners cooperatively maintain the blockchain system with the P2P network. In this paper, we adopt the system model composed of four parts, i.e., distributed ledger, consensus mechanism and mining, smart contract platform, and application, which is shown in Fig. 1.

Distributed Ledger: The distributed ledger is a decentralized database that records all blockchain data in a standard format and is

Table 1
Comparison with existing surveys.

	Investigation the Latest Works	Investigation of On-Chain Payment	Investigation of Privacy-Preservation of Off-Chain Channel	Investigation of Privacy-Preservation of Smart Contract
[20]	No	Yes	No	Yes
[21]	No	Yes	No	No
[22]	No	Yes	No	No
[23]	No	Yes	No	Yes
[24]	No	Yes	No	No
This work	Yes	Yes	Yes	Yes

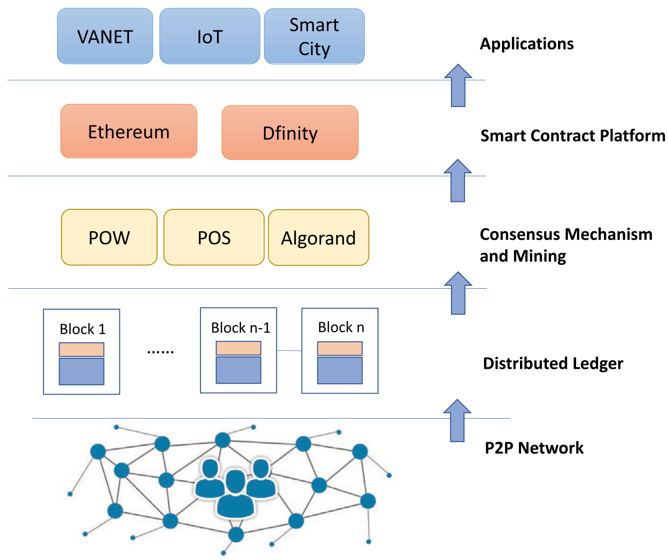

Fig. 1. Blockchain architecture.

maintained by all miners. It includes a series of blocks that are connected in the chain using the hash function. The blocks are organized chronologically, and each block is identified by its hash value which is called block address. Fig. 2 presents a typical block structure consisting of a block header and a block body. The block header includes the current version number, the hash value of the previous block (i.e., block address), its own block address, the Merkle root hash, and the timestamp when the block is created. For Proof of Work (PoW)-based blockchain, it contains a nonce to prove that the block is correctly generated. The block body includes all the confirmed transactions, which are permanently recorded in the blockchain. All transactions are organized by Merkle Tree [25] for efficient transaction querying and verification.

The miners are responsible for maintaining the distributed ledger. They can access the data in the ledger and write data into it. However, before a piece of data is recorded into this ledger, its validity must be verified and confirmed via the consensus mechanism. A user can access the data, but only write the data into blockchain with the assistance of a miner.

Consensus Mechanism and Mining: A consensus mechanism is a fault-tolerant mechanism that enables multiple parties to achieve the necessary agreement on a single data value or a single state of the network [26]. It provides the core functionality to maintain the originality, consistency, and order of the blockchain data across the network. Mining refers to the process that miners reach a consensus on a newly created block via blockchain, which provides liveness and safety.

Generally, a blockchain system such as Bitcoin is secure as its consensus model [22]. The security of consensus relies on the premise of honest-majority, namely, the majority of consensus voting power is honest [27]. Some blockchains, such as Bitcoin [28] and Ethereum [29], include an incentive mechanism to motivate miners to create new blocks. The incentive helps improve the durability of the blockchain and, based on game theory, enhance the security of the blockchain.

Smart Contract Platform: A smart contract is a computer program running on the blockchain, which extends the functionality of the blockchain and enriches the application of the blockchain [30]. There are several definitions of smart contracts. For example, Szab [80] creatively proposed that "smart contract is a computable transaction protocol to execute contract terms"; Ethereum's smart contract [29] is a digital asset control program based on blockchain. In a narrow sense, a smart contract is a program code that involves business logic, algorithms, and program complex relationships among people, legal agreements and networks. In a broad sense, a smart contract is a kind of computer protocol that can realize self-execution and self-verification after its deployment.

The operation of smart contracts includes three procedures: contract generation, contract publishing, and contract execution. During contract generation, the contract participants in contract execution will negotiate to clarify the rights and obligations of the parties, determine the standard contract text, and then program them into a smart contract program. Usually, the contract program needs auditing for secure execution. In contract publishing, the contract generator signs the contract and requests a miner to record the signed contract into the blockchain. The contract execution is based on an event-triggered mechanism based on blockchain, which contains transaction processing and preservation mechanisms and is a complete state machine. To be specific, the external nodes can interact with a smart contract program by sending particular transactions. The transactions can change the status of the contract. All miners monitor the status, and once detecting its change, they execute the smart contract based on its design.

2.1.2. Applications based on blockchain

The distributed ledger and smart contract platform enable users to run various applications on top of the blockchain. Its decentralization greatly enhances the resistance to the risk of a single point of failure and security risks due to distrusted centralized parties. Therefore, blockchain-based applications quickly attract continuous attention in academia and industry and show a promising application prospect. To better illustrate the potential applications of blockchain, we here list several typical applications of blockchain.

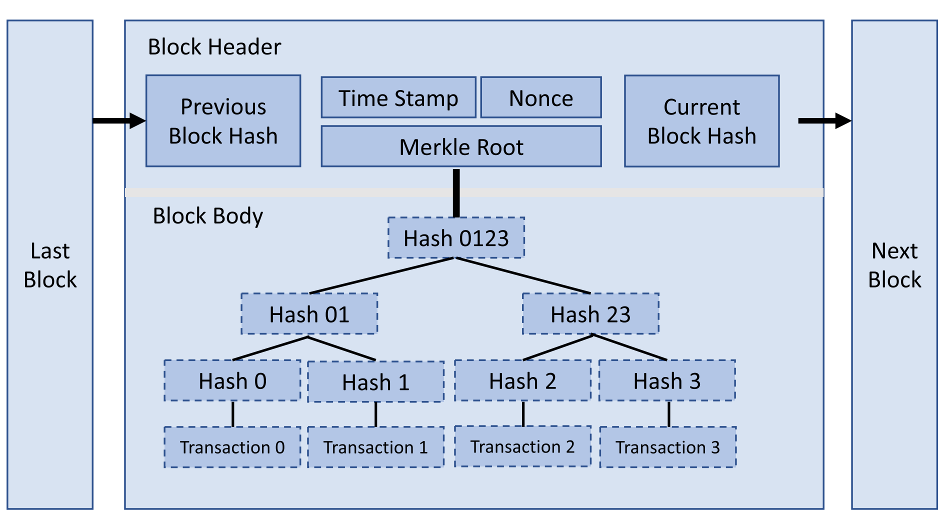

Fig. 2. Structure of a block.

Financial applications: The emergence of blockchain results in a significant change in the business model of finance [31–33]. Blockchain can generate trust spontaneously in the decentralized system and can establish a financial market without a trusted centralized party, which is a revolutionary transformation for the business model of intermediaries such as payment services with third-party. Due to its transparency and irreversibility, blockchain technology is very suitable for financial applications such as cryptocurrency and P2P lending [34]. The use of blockchain smart contracts and alternative features can greatly reduce costs and improve efficiency, avoid cumbersome centralized capital settlement process and achieve convenient and fast financial product transactions, which is currently an important driving force for research and investigation into blockchain from big companies.

Digital Voting: Voting is a representative application of blockchain in political affairs [35–37]. It can achieve political elections and corporate shareholder votes at a lower cost. Blockchain-based voting can also be used for games, forecasting markets, and recommendations.

Other real-world applications: Blockchain achieves decentralization, data immutability, and trust. These features make the blockchain widely applicable to various types of data notarization and audit scenarios [38]. For example, blockchains can permanently and safely store all kinds of licenses, registration forms, certificates, certifications and records issued by government agencies and can easily prove the existence and certain degree of authenticity of certain data at any time. Blockchain can also be applied into many decentralized scenarios, such as clock synchronization scheme [39–41], mobile crowdsourcing [42], searchable encryption [43–45], secure storage system [46–49], energy trading [50], etc. [51,52].

2.2. Unique characteristics

Based on the system model and applications of the blockchain, we analyze the unique characteristics of the blockchain in this section.

2.2.1. Decentralization and autonomy

Decentralization and autonomy refer to the system that contains no centralized party for maintenance and management. Each node can access and verify the entire database and its complete history and jointly maintain the evolution of the system. The underlying consensus mechanism ensures blockchain's security and regular operation. The decentralization and autonomy of the blockchain help resist the risk of a single point of failure and privacy leakage due to distrusted authorities. However, it still incurs more privacy risks because the adversary can harm a group of miners more easily (Fig. 1).

2.2.2. Openness

A permissionless blockchain is an open system that every node can join and leave network freely. The openness makes it possible to recruit numerous miners for blockchain maintenance and also allow adversaries more opportunities to intrude into the blockchain system.

2.2.3. Non-repudiation

The non-repudiation of blockchain refers to that (i) no one can deny transaction contents created by himself; (ii) no one can repudiate the transaction time generated by himself. Due to the characteristic of non-repudiation, as long as a transaction exists in the blockchain, it must be initiated by its signer itself, and the node cannot deny that it has published the transaction.

2.2.4. Verifiability and immutability

Verifiability and immutability mean that the validity of each transaction in the blockchain can be verified and cannot be modified or removed from the blockchain. Since the blocks in which transactions are recorded are confirmed by all miners via the consensus mechanism, invalid transactions will not be recorded in the blockchain, and any modification on data in the blockchain will be denied unless the adversary compromises the whole system. Additionally, blocks are organized in the form of the chain using the hash function, which makes any modification on the data easily detected. This characteristic benefits security but also results in the problem that sensitive data cannot be removed from the blockchain.

2.2.5. Transparency

Data in the permissionless blockchain are transparent to all miners, and users can conveniently access on-chain data by querying miners. The transparency enhances data immutability and verifiability because all nodes can detect illegal data modification and illegal data. Nevertheless, privacy leakage due to transparency has become a crucial issue that dramatically limits the application of the blockchain.

3. Privacy threats and requirements on privacy preservation in the blockchain

Based on the proposed system model and the analyzed characteristics of blockchain, we further define the security model and analyze the privacy issues in the blockchain. Besides, we summarize the potential threats to privacy in the blockchain based on which we propose a series of requirements on privacy preservation in the blockchain.

3.1. Privacy issues in blockchain

3.1.1. Transaction privacy

Data in the blockchain is public to all, thus keeping the information in sync and reaching consensus among distributed nodes, which results in privacy risks. For one thing, transactions may contain sensitive information about their owner. With the popularity of blockchain in various scenarios, such as Mobile CrowdSourcing (MCS) and the IoT, direct privacy leakage due to transaction exposure becomes a crucial issue. Additionally, the disclosure of transaction content may also face indirect privacy leakage. For example, by analyzing the transaction graph, the adversary can obtain the correlation between transaction addresses and infer the user's real identity from extra data, which seriously threatens the users' privacy. Therefore, the blockchain system should pay more attention to transaction privacy issues and improve the level of privacy protection.

3.1.2. Privacy of smart contract

Smart contracts inherit some undesirable blockchain properties. The general smart contract requires every miner to execute every step of every smart contract, which needs the code and data of every contract to be public. Private information can not be preserved during the validation of state transitions via consensus. Therefore, existing smart contract systems thus lack data confidentiality (e.g., auction bids, financial transactions), which bring serious privacy problems.

3.2. Threat model

In our paper, we adopt the threat model, that is, the blockchain system contains no fully trusted party. Both miners and users are rational and behave based on the information recorded in the blockchain and their own benefits. We consider an adversary that can compromise an arbitrary set of miners or users. However, it cannot break the security of the blockchain system. Aiming at disclosing the privacy mentioned above, we summarize several potential attacks that may be conducted by the adversary as follows.

3.2.1. De-anonymization and tracking

In the blockchain, users usually use hash values of randomly selected public keys as identifiers to hide their real identities. However, it is possible to disclose the users' real identities or track their activities by analyzing their transactions. Typically, Reid et al. [53] analyzed the input and output relationships in the transaction network by constructing a payment linkage graph and then aggregated multiple inputs to a single address to indicate that multiple-input transactions were generally initiated by the same owner signature. The user's public key and the information provided by the relevant website pose a threat to the user's identity privacy. For example, if a user purchases goods online using Bitcoin, the online store could access details such as the user's email address, shipping address, IP address, etc. [54].

However, it is not enough to only guarantee user identity in the blockchain. For one thing, the blockchain is widely used in various applications, such as IoT, MCS, VANET, etc. In these systems, a transaction usually contains more information than just the number of coins. In this case, attackers are able to infer the real identity of transaction generators by analyzing transaction content with additional information. For another, attackers can analyze the relationship between different transactions to obtain the relationship between them. In this way, attackers can track the activities of a single user.

Apart from identity inference with extra knowledge, more methods of identity inference and user transaction tracking are proposed. Meiklejohn et al. [55] used a clustering heuristic algorithm to cluster the addresses of the same user. They effectively marked each other's public key as a service provider by conducting actual transactions with a number of service provider websites and combined the addresses published in various forums and websites. Therefore, the service provider could be classified according to the marked public key, including the exchange, the mining pool, and so on. According to the service provider's public ledger information, the association of addresses in the ledger could be obtained to reduce user anonymity.

Ron et al. [15] analyzed the transaction relationship of the bitcoin system through the Union-Find algorithm and associated each public key with a different address for 3730218 different public keys in the ledger. They finally obtained 2,460,814 different owners and speculated that there are many different exchanges, mining pools, etc. Koshy et al. [56] created a mapping from bitcoin addresses to IP addresses by analyzing bitcoin transaction information. By creating a bitcoin wallet CoinSeer with a data collection function, they collected and analyzed five months of transaction data, classified different transaction relay modes, and finally analyzed three abnormal relay modes. They discovered the transaction initiating nodes and created the mapping of addresses to IP addresses of bitcoins. This indicated that certain bitcoin address sets could only be de-anonymized by observing the transaction relay mode.

3.2.2. *Transaction flow leakage*

In the Bitcoin system, all transactions are open and transparent, and users can get full transaction content. The chain structure of the blockchain and the Merkle tree structure makes every transaction of the system traceable. Bitcoin uses the Unexpected Transaction Output (UTXO) transaction mode. A transaction can have multiple inputs and multiple outputs. The current transaction input is the output of the previous transaction, and the current transaction output is the input for the next transaction. According to the correlation of the transaction address, an attacker can track the transaction and obtain the monetary flow. Some users do not want to disclose the transaction content to protect the transaction data.

At the bitcoin trading website, detailed information about transactions associated with the public key address can be obtained based on the user's public key. Reid et al. [53] obtained the public key address of the user through the website such as Bitcoin Forum and Twitter, tracked the source and usage of the user's funds, and calculated the user's balance combining the knowledge of the monetary flow of the stolen address before and after the theft. Ober et al. [66] analyzed the bitcoin transaction topology map and observed the relationship between the number of active entities and the bitcoin exchange rate. Based on their study, the increase in the exchange rate would increase the number of active entities. According to the transaction relationship graph between the addresses, the authors discovered the quantitative relationship between the bitcoin trading system's dormant bitcoin changes in different periods.

3.3. *Requirements of blockchain privacy preservation*

In this section, we analyze the requirements of privacy preservation schemes for the blockchain.

Transaction confidentiality: Transaction confidentiality means that transaction content cannot be accessed by unauthorized entities. Permissionless blockchain usually allows everyone to access transactions in the blockchain. Nonetheless, the current blockchain is widely used in various systems where transactions are likely to contain sensitive data and lead to direct privacy leakage. For example, the transaction records of users' shopping can reflect the user's consumption level, living status, etc. In practice, users wish to have the least disclosure of transactions and account information in the blockchain system. Therefore, it is necessary to take measures to limit access to blockchain data.

Anonymity: As explained in Ref. [57], anonymity means that the subject is not identifiable in a group of subjects, i.e., an anonymity set. In the blockchain, we refer to anonymity as the fact that the adversary cannot distinguish the particular individual from a set of real identities, whose size depends on the privacy preservation method. Anonymity is the basic requirement for identity privacy preservation, while the blockchain's transparency brings about many privacy issues in some scenarios, especially in the financial field. Considering the increased attention of users to privacy, especially identity privacy, a practical privacy preservation scheme for permissionless should first achieve anonymity.

Transaction unlinkability: Different from the anonymity defined above, users also require that the transactions related to themselves cannot be linked. A blockchain address is a pseudonym used by a user in the blockchain system. It usually works as the input or output account of a transaction. The address in the blockchain system is generated by the user, which is independent of the user identity information. The user creates and uses the address. Third-party participation is required. Therefore, the blockchain address has better anonymity than the traditional account(such as bank card number). However, users may leak some sensitivity when using the blockchain address to participate in the blockchain service. Information such as the propagation trajectory of blockchain transactions at the network layer may be used to guess the true identity of the blockchain address. So unlinkability is an important factor that we should consider.

Efficiency: The blockchain itself confronts severe efficiency problems like low throughput, while smart contract based on blockchain suffers from high computation overhead. Therefore, the privacy preservation schemes should not cause the efficiency degradation of the blockchain system. A practical privacy preservation should achieve efficiency in communication, computation, and storage, and it is significant to ensure that the efficiency reaches an acceptable level when designing privacy preservation schemes.

Fairness: Fairness in financial system measures the health of the system. When it is specific to the blockchain, it means that the interests of either party will not be damaged in the blockchain transaction. It is significant for users to believe that the privacy-preserving blockchain system they use can ensure fairness, so it is an essential requirement that should be listed here.

Compatibility: The compatibility measures the capacity of the methods applied in different systems. AS the most famous blockchain system, Bitcoin has been treated as a system that needs to be compatible with many projects, which also brings more users' acceptance to their works. Thus, whether the method can be compatible with bitcoin should a factor that we need to consider.

3.4. *Criteria for evaluating schemes*

In this part, we will list a set of criteria and make a comprehensive comparison of privacy preservation techniques used in the blockchain. Different type of techniques raises different features that need to be compared. We list the criteria below by which we will then evaluate the approaches we discussed in section 4.

Privacy protection: The privacy includes the anonymity of participants, the number of payment transactions, the input and the state of the smart contract. The concrete meaning depends on the categories of the privacy preservation methods.

Compatibility: Compatibility refers to the capacity of the approaches to be applied to different systems. Whether it is compatible with Bitcoin or Ethereum, blockchain privacy preservation methods affect the user's acceptance of this approach.

Protection of coin theft: For mixing services, the funds of payments held by users need to be protected securely while using a mixer for anonymous operations.

Requirement of centralized party: For mixing services, the central third party included in the scheme will bring some security problems, and other approaches can avoid this risk. So, this is a criterion that we need to consider.

Requirement of mixing fee: The mixing fee charged during the mixing process will decrease the user experience without any doubt, which is also an important criterion.

Anonymity set: Anonymity set refers to the size of space from which the party's identity cannot be distinguished, which measures the degree of the identity privacy preservation of the approach. The anonymity set varies when it comes to different approaches.

Requirement for trusted setup: For crypto-based privacy-preservation techniques, such as zero-knowledge proof, may require a setup process whose security needs to be protected by a trusted execution environment or secure multiparty computation technique. The compromise of the trusted setup will ruin the security of the privacy preservation system of the blockchain.

Transaction size: Transaction size refers to the average size of each transaction in the blockchain system. The larger the transaction size, the lower the blockchain performance. The use of crypto-based methods can easily incur a cumbersome transaction with additional protection, so the privacy guarantee and performance need to be balanced in practical use.

Functionality: For the methods of protecting off-chain channel privacy, functionality means that the type of function of the underlying system architecture.

Channel direction: For the methods to protect off-chain channel privacy, channel direction refers to the support direction of the approaches. Payments can be conducted unidirectionally from payer to payee in a unidirectional approach while the bidirectional method supports payments to each other in a single channel.

Parties executing smart contract: For the methods to protect off-chain channel privacy, the number and type of the parties that run the smart contract vary with the schemes. The more the parties need to execute the contract, the more computing overhead is in progress, while it will also result in a low risk of a single point of failure and relatively high system stability.

4. Methods of protecting privacy

In this section, we categorize all privacy preservation methods into two categories, i.e., transaction-related privacy preservation and smart contract related privacy preservation. Then we comprehensively analyze their advantages and disadvantages with the proposed evaluation criteria.

4.1. On-chain transaction privacy

4.1.1. Mixing services

Transactions in permissionless blockchain are public to all. Therefore, an attacker can look up transaction content (including the transaction amount and transaction addresses of both payer and beneficiary) and infer the implicative information in each transaction. Therefore, the openness and transparency of permissionless blockchain will harm the privacy of users. One of the prominent solutions to this problem is the mixing service. Mixing service was first proposed by Chaum [58] in communications, which have been integrated into the blockchain these years to alleviate the risk of de-anonymization by obfuscating the input and output of transactions. Its main idea is to allow multiple users to jointly form a single transaction with multiple inputs and outputs. In this way, an attacker cannot link the transaction input to its corresponding output. Existing mixing services can be divided into centralized mixing and decentralized mixing based on whether a third party is needed. In this subsection, we analyze the mixing services of both two categories.

a) Centralized mixing

In centralized mixing, the centralized party called the mix server is responsible for generating the transaction that contains the inputs and outputs of all users. A simple example of centralized mixing is that all users transfer the bitcoins to a mixing server, and the server then transfers the bitcoins to the corresponding beneficiaries. The structure of a typical centralized mixing service is shown in Fig. 3. Generally, the user needs to pay a certain amount of coins to the mix server as a reward. This design is somehow effective to ensure anonymity. However, it faces a crucial coin theft problem because users can hardly ensure the authenticity of the untrusted central service. Therefore, it is not practical to use a centralized server to mix coins.

To mitigate the coin theft problem, Bonneau proposed the Mixcoin that was compatible with the Bitcoin [59]. Mixcoin achieved anonymous payment with the assistance of the responsible mixer, which was operated as follows. When the user sent a bitcoin to the mixer, he also got a signed warranty from the mix server, which served as a commitment to the fairness of the exchange. If the mixing server broke the mixing protocol, the sender could use the warranty to disclose the malicious behavior to reduce its reputation. Obviously, Mixcoin was effective to solve the coin theft problem only when the server was rational. Besides, the server was well aware of the transaction inputs and their corresponding outputs, and thus the mixing server could easily break the anonymity. Valenta and Rowan further optimized the centralized mixing service by using blind signature technology and designed Blindcoin [60]. Blindcoin could ensure that the third party cannot establish a link between the input and output addresses of transactions while providing mixing services. It could prevent the third party from disclosing the transaction relationship of users and achieve full anonymity. However, as an extension to Mixcoin, Blindcoin also suffered from the coin theft problem and could only provide a limited security guarantee.

Unlike the above schemes based on a single fixed mixer, Dash [61] leveraged a set of mixer nodes called master nodes to offer a mixing service. It was a digital currency platform with privacy preservation. In order to improve the anonymity of the transaction, Dash allowed a user to randomly select several master nodes for coin mixing, thus keeping the association between addresses invisible. In Dash, master nodes must pay 1,000 Dash coins as a deposit in advance to provide mixing services,

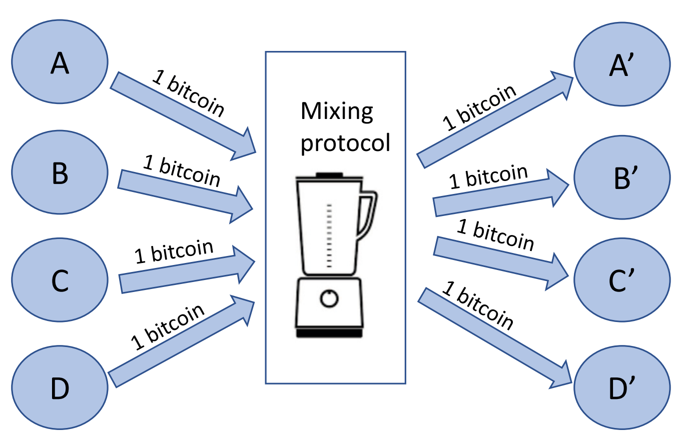

Fig. 3. Centralized mixing service.

which increased the cost of protocol violations and mitigated the coin theft problem. Similar to Mixcoin, Dash only supported fixed-denomination payments and could not resist privacy leaks due to inner attacks by malicious master nodes. Besides, the number of mixing participants was limited, which limited in turn its application in the real world.

Coinswap [62] proposed by Maxwell was the first work to solve the coin theft problem. It utilized escrow transactions and fair exchange protocols to provide coin mixing service through an intermediary. The payment transactions took the form of escrow transactions and used two escrow protocols to guarantee that the payee received funds when and only if the mixer received funds from the payer, all of which were protected by a fair exchange protocol. However, the multiple rounds of interactions between client and intermediary limited its performance in practice.

Heilman et al. [63] proposed an anonymous payment scheme that includes third parties. It offered two anonymous payment solutions, i.e., on-chain solution and off-chain solution. The on-chain solution introduced an untrusted intermediary between all payers and beneficiaries. Set anonymity was provided during each period during which the protocol runs. That is, although the blockchain publicly displayed the collection of payers and beneficiaries at a specific time, no one could tell the payer that the payee had paid. The off-chain solution adopted a new payment method named the micropayment channel networks. This micropayment channel network is paid through the pre-established path of the connected user. Therefore, the users participating in the path would know the transaction details, including the encrypted identity of the sender and the receiver. Introducing a semi-trusted third party could provide anonymity against malicious users while protecting user privacy from the outside world, but also result in an internal anonymity problem as most mixing services face.

Either of the above schemes cannot achieve payment fairness, or they could not well support anonymity. Besides, few of them could solve the privacy leakage problem due to inner attackers. Motivated by these challenges, Heilman et al. proposed a hybrid system named Tumblebit [64], which was built upon the Bitcoin system and thus achieved better security. Tumblebit combined the RSA puzzle with fair exchange techniques to build an anonymous and secure Bitcoin-based payment system via an untrusted intermediary, i.e., the tumble. The on-chain bitcoin payments were replaced with off-chain puzzle solving, which meant that the beneficiaries should have the solution to the puzzle instead of only a specific secret related to the address. Two escrow transactions would be generated during one payment to ensure fairness. The RSA puzzle was generated and solved during interactions between the payer, the tumble, and the beneficiary with the fair exchanged protocol to avoid violation. The anonymity of Tumblebit guaranteed that no one could deduce the transaction linkability. However, if the tumble colluded with the beneficiaries, it was easy to learn the real identity of the payer. Besides, Tumberbit supported neither payment values hiding nor bidirectional payment channels, affecting its availability in practice.

b) Decentralized mixing

Centralized mixing services mainly rely on a trusted or semi-trusted third party to mix the transaction sets of multiple users and output them to the corresponding addresses so that attackers cannot link the input and output addresses of the transaction. Effective as they are, they suffer from the risk of a single point of failure like most centralized systems. As a result, the alternative approach, i.e., decentralized mixing, have been quickly explored afterwards, which benefits users since it needs no mixing fees. The structure of a typical decentralized mixing model is shown in Fig. 4.

CoinJoin [65], proposed by Maxwell in 2013, was one of the first decentralized mixing services, where users could mix their coins in a self-organized way instead of relying on a third party. At the beginning of the hybrid era, a negotiation process would be conducted among a group of payers to determine to whom they wish to make the joint payment.

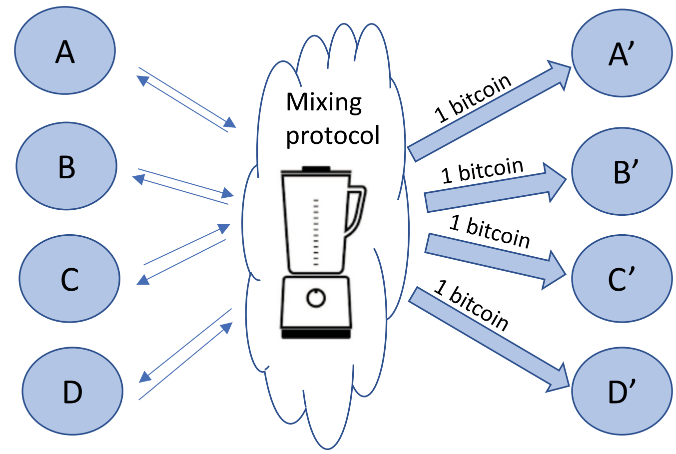

Fig. 4. Decentralized mixing service.

Then, a transaction containing all input/output pairs was generated and checked by users to ensure that their payment destination was properly encapsulated. It also achieved obfuscation by shuffling the addresses. If the transaction was verified by all the payers, they would jointly sign the transaction and eventually published it via the blockchain. Compared with centralized methods, CoinJoin significantly reduced the risk of deduction of transaction linkage due to outer/inner attackers and eliminated the problem of coin theft. However, CoinJoin still has some shortcomings. During the negotiation, the users participating in the coin mixing might discover information about other clients. In addition, CoinJoin was vulnerable to the Denial of Service (DoS) attack. Specifically, if any user in the mixing set was unavailable or abnormal, the entire mixing process would fail. As a result, its availability is low.

CoinShuffle [66] utilized a novel accountable anonymous group communication protocol named Dissent [67] to provide inner anonymity. All users in the mixing set conducted nested encryption of the outputs in a predetermined order using the public keys of other users. They shuffled the output addresses in order, and then the output address list was broadcasted to all participants. Each user checked whether the transaction contained his correct destinations and signed the transaction. The final transaction would be published to the blockchain once all signatures were gathered. CoinShuffle ensured that no one could get the connection between the transactions even for the participants with the absence of a centralized party. However, all participants needed to be online during the mixing process. Similar to CoinJoin, CoinShuffle were also vulnerable to the DoS attack.

CoinParty [68] was a distributed hybrid technology based on Secure Multi-Party Computation (SMC). SMC enabled a group of parties to jointly generate a shared address without leaking their secret input. The new address would be set as a beneficiary address, and a threshold of signatures is needed to redeem the coin. However, its security needed only to be guaranteed when more than 2/3 of the parties are honest, which did not hold in most scenarios.

Dining Cryptographers network (DC-net) protocol was proposed by Chaum [69] for mixing data senders' identities to achieve anonymous communication, which supported multiple senders [70]. DiceMix [71] is a general decentralized hybrid method proposed on the basis of the original DC-net protocol to protect the anonymity of the sender. By using DC-net, it could break the connection between the payer address and beneficiary addresses. Besides, it significantly reduced the communication overhead of the DC-net, and meanwhile, it could also resist malicious peers. In addition, based on the ideas of CoinJoin and DiceMix, CoinShuffle++ deviated from DiceMix was a decentralized mixing protocol that was compatible with Bitcoin, which significantly reduced the communication bandwidth consumption and improved performance compared with original CoinShuffle. However, the anonymity set was still relatively limited.

In Table 2, we compare all the efforts of mixing services based on the criteria listed in section 3 to give a comprehensive overview.

4.1.2. Ring signature & confidential transactions

The ring signature was originally described in Ref. [72]. It was a special group signature through which a user could anonymously sign on behalf of a group of users, including the actual signer. Compared with the group signature, there was no trusted center and group establishment process. For the verifier, the signer was anonymous, and the verifier could not analyze its specific identity. A ring signature algorithm must satisfy the following properties: (i) Unconditional Anonymity: An attacker cannot determine which member of the ring is generated by the attacker. Even if the ring member private key is obtained, the probability does not exceed 1/n. (ii) Correctness: The signature must be verified by everyone. (iii) Unforgeability: Other members of the ring could not forge the signature of the real signer. An external attacker cannot forge a signature for the message m even if he obtains a valid signature. Ring signature has a variety of applications in scenarios where the signer's identity needs to be preserved, such as anonymous identity verification in the ad-hoc group [73] and cryptocurrency [74].

The ring signature is first applied to CryptoNote to hide the origin of transactions [74]. CryptoNote is an evolution of Bitcoin, which can protect the identity privacy of both payer and payee of a transaction. In CryptoNote, a transaction is signed and verified with a ring signature, and verifiers can only ensure that its signer belongs to a specific user-set but cannot distinguish its real identity. For its payee, it can create a pair of unique one-time private and public key pairs with some randomness chosen by the payer and the payee's public address. To be specific, a payer generates a one-time key for each transaction, and only the payee can recover the corresponding private key. CryptoNote implements that no third party can determine whether two transactions are sent to the same beneficiary, which results in the external invisibility of the beneficiary's address. To prevent double-spending attacks due to unidentifiable payers, it leverages traceable ring signature [75] to track the sender who tries to sign multiple transactions twice to use the same coin.

Inspired by CryptoNote, several cryptocurrencies were developed based on a similar idea, the most famous of which was Monero [76]. CryptoNote was based on Confidential Transaction proposed by Maxwell [77], which employed a commitment scheme to hide the amount of the transaction. Monero leveraged the ring signature and one-time unique address in CryptoNote to extend the Confidential Transactions [77] to Ring Confidential Transactions (RingCT) [76] for transaction confidentiality. RingCT introduced a Multilayered Linkable Spontaneous Anonymous Group signature to combine the Pedersen Commitment with ring signatures. Owing to RomgCT, Monero achieved transaction irrelevance and hidden transaction amounts. Specifically, it used ring signatures and one-time addresses to break off the link between the input address and the output address in each transaction, and used Confidential Transactions to hide the amount. However, a recent study shows that its anonymity can be broken probabilistically through deduction. Therefore, Monero cannot well ensure anonymity [78].

The original RingCT suffers from a large transaction size, which is linear to the number of input addresses in an anonymity set. To reduce the size of the original protocol, RingCT 2.0 was proposed based on the Pedersen commitment, linkable ring signature, and accumulator with a one-way domain [79]. The accumulator can provide anonymity and transaction confidentiality, and meanwhile, it significantly shortens the size of each block. Since its construction fits perfectly into RingCT definition, it is compatible with Monero. Recently, RingCT 3.0 was proposed [80], it removes the trusted setup assumption and significantly reduces the ring signature size, which makes it a candidate to be next-generation technology used in Monero.

4.1.3. Non-interactive zero-knowledge proof

Zero-knowledge proof, first introduced in early 1980 [81], is a powerful technology that can be applied to privacy protection. A zero-knowledge proof is a method by which the prover can convince a verifier that a particular assertion is correct without leaking any useful information. The security guarantees are (i) Completeness: If the statement is true, and both users follow the rules properly, then the verifier would be convinced that the statement is true. (ii) Soundness: If the statement is false, the prover cannot convince the verified that the statement is true under any circumstances. (iii) Zero-knowledge: Nothing else should be leaked to the verifier. Both of the above situations need to hold with an overwhelming probability.

Zerocoin[82], proposed by Miers et al., was the first privacy-preserving payment scheme based on the Zero-Knowledge Proof of Knowledge (ZKPoK) [83]. It was an extension of Bitcoin, allowing users to cast a bitcoin into a zerocoin for trading and redeem a zerocoin back into a bitcoin. When using zerocoins for trading, other users cannot obtain any trading information and can only check whether the zerocoin has been spent, which can break the linkability of transactions. Zerocoin employs ZKPoK to prove that a zerocoin originates from an unspent bitcoin, and it is computationally infeasible for any adversary to trace the zerocoin to its corresponding bitcoin. Based on Zerocoin, an Enhanced ZeroCoin (EZC) [84] was proposed, which is superior to Zerocoin since it can hide transaction amount and address balance, which is not supported by Zerocoin. Besides, a user must convert a zerocoin back to a bitcoin for spending, and then it only supports the conversion of a single bitcoin. Differently, EZC enables the consumption of zerocoins without converting them back to

Table 2
Comparison of mixing services.

Proposals	Privacy Protection	Compatibility	Protection of Coin theft	Requirement of Centralized Party	Requirement of Mixing Fee
Untrusted central mixing service	No	Compatible with Bitcoin	No	Yes	Yes
Mixcoin [59]	External anonymity	Compatible with Bitcoin	Accountable	Yes	Yes
Blindcoin [60]	External/internal anonymity	Compatible with Bitcoin	Accountable	Yes	Yes
Dash [61]	External anonymity	Compatible with Bitcoin	Accountable	Yes	Yes
Coinswap [62]	External anonymity	Compatible with Bitcoin	Yes	Yes	Yes
Heilman's work [63]	External anonymity	Not compatible with Bitcoin	Yes	Yes	Yes
Tumblebit [64]	External/internal anonymity	Compatible with Bitcoin	Yes	Yes	Yes
CoinJoin [65]	External anonymity	Compatible with Bitcoin	Yes	No	No
CoinShuffle [66]	External/internal anonymity	Compatible with Bitcoin	Yes	No	No
CoinParty [68]	External/internal anonymity	Compatible with Bitcoin	Yes if 2/3 honest	No	No
CoinShuffle++ [71]	External/internal anonymity	Compatible with Bitcoin	Yes	No	No

bitcoins and allows conversion of multi-valued zerocoins with values never revealed to any other party except for the payer and the beneficiary. Compared with Zerocoin, EZC achieves lower communication overhead. In summary, though Zerocoin effectively realizes anonymity, it can only mint and redeem fixed-denomination currency. Besides, because of the large proof size of the ZKPoK scheme, Zerocoin introduces additional blockchain storage and computing resources.

To overcome the weaknesses of Zerocoin, Miers et al. further proposed Zerocash [85]. This follow-up project of Zerocoin [82] was a full-fledged ledger-based digital currency with strong privacy guarantees that uses Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARKs) [86] as the core technology. Compared with Zerocoin, Zerocash guarantees the confidentiality of the transaction amount and supports the payment of any denomination. A user can mint coins of different denominations into multiple coins of equal amount, each with its own amount, serial number, and so on. During the minting process of coins, a user needs to generate a commitment and adds it to the common commitment list. To transfer coins to the beneficiary, the user encrypts the transaction content (i.e., amount and the beneficiary's address) with the public key to the beneficiary and broadcasts the encrypted transaction to the entire network. After the beneficiary obtains the transaction content with the private key, it generates the serial number for these coins. When using zk-SNARKs to verify a transaction, a miner only needs to confirm that the validity of proofs provided by the transaction initiator. However, the miner is unable to distinguish the corresponding commitment, thus ensuring anonymity. Each coin is identified with a unique one-time serial number, which can effectively prevent double-spending attacks. The utilization of zk-SNARKs remarkably improves the performance by reducing the proof size and verification time. Despite the excellent performance in privacy preservation and efficiency of Zerocash, its security requires a trusted setup process that determines the parameters of zk-SNARKs. If the adversary compromises this process, it can get the master for coin generation and break the security and privacy guarantees of Zerocash.

The shortcomings of Zerocash in performance and security motivate the emergence of more zero-knowledge proof-based privacy preservation schemes. Bulletproofs [87] was a powerful scheme that provides short and aggregated range proofs, which remarkably improves the performance of zk-SNARKs. It dramatically reduces the size of existing range proofs technologies and supports proof aggregation, allowing users to prove multiple commitments with a single proof. It is possible for multiple parties to jointly generate a single proof without revealing inputs via secure multi-party computation. Bulletproofs is currently the most efficient range proof that is promising to form a variety of decentralized cryptocurrencies and applications [88,89].

In Table 3, we compare all efforts of crypto-based techniques based on criteria listed in section 3 in order to give a comprehensive overview.

4.2. Privacy preservation for off-chain payment channel

The off-chain payment channel was introduced by Spilman [90] and has flourished as a promising approach to reduce payment delay and transfer fees in the on-chain payment system. In a nutshell, a payment channel enables a payer and a beneficiary to establish a payment contract upfront through an online transaction that escrows funds temporarily, after which the payer and the beneficiary can keep track of the funds they owe each other and then locally agree on the new distribution of the deposit balance to update the contract. The payment channel avoids recording payment details on the blockchain, and the final payments can be made instantaneously via a closing transaction.

Heilman's work in 2014 [63] was a pioneer in considering anonymity in the off-chain payment channel. A user willing to make a payment first needs to establish a payment path. Its weakness is that all users included in the path will obtain the identity information of the payer and the beneficiary. An improved approach includes a semi-honest intermediary to protect privacy from outer attackers. However, the introduced intermediary can link their transactions. Besides, it is not compatible with Bitcoin.

Green et al. proposed an anonymous payment channel scheme called Blot [91], in which users conducted most off-chain transactions based on Bitcoin-like cryptocurrencies, such as Zerocash. Blot offers three modes of off-chain payment: unidirectional payment channel, bidirectional payment channel, and indirect payment channel. Transactions between users can be made directly through a secure off-chain channel or with the assistance of untrusted third parties. Bolt provides a way that a payer can create an anonymous direct channel even if the beneficiary does not know the identity of the payer. The indirect payment channel uses the blind signature technology and zero-knowledge proof to prevent the third parties from obtaining the user's transaction information. Besides, it utilizes the compact e-cash paradigm described in Ref. [92] to guarantee a constant transaction size regardless of its volume. However, the third party's failures can cause monetary loss, and the strong privacy protection against an intermediary payment channel hub relies on the privacy property of the cryptocurrency it is built upon. Besides, a payer requires an existing long-lived relationship with an intermediate payment hub or the beneficiary for privacy-preserving payment, which may not be available in practice and cannot work well for those with limited bandwidth.

Tumblebit [64] is compatible with classic Bitcoin, allowing for anonymous payment channels between different users, as already discussed in section 4.1. It does not support any denomination and payment value hiding. Besides, the collusion of the payee and the tumble will break the anonymity of a payer.

To eliminate the limit of throughput and the long-lived financial connections between parties, a privacy-preserving Payment-Channel Network (PCN) with multi-hop payments [93] was proposed, which

Table 3
Comparison of crypto-based techniques.

Proposals	Privacy Protection	Compatibility	Anonymity Set	Requirement for Trusted Setup	Transaction Size
CryptoNote [74]	Hiding addresses of participants	Not compatible with Bitcoin	Small	No	Small
Monero with RingCT 1.0 [76]	Hiding transaction amount, addresses of participants	Not compatible with Bitcoin	Small	Yes	Large
RingCT 2.0 [79]	Hiding transaction amount, addresses of participants	Not compatible with Bitcoin	Small	Yes	Middle
RingCT 3.0 [80]	Hiding transaction amount, addresses of participants	Not compatible with Bitcoin	Small	No	Small
Zerocoin [82]	Hiding addresses of participants	Not compatible with Bitcoin	Large	Yes	Large
EZC [84]	Hiding transaction amount, addresses of participants	Not compatible with Bitcoin	Large	No	Large
Zerocash [85]	Hiding transaction amount, addresses of participants	Not compatible with Bitcoin	Large	Yes	Middle
Bulletproofs based work [88,89]	Hiding transaction amount, addresses of participants	Depends on protocols	Large	No	Small

allowed for payments between users that do not have a direct payment channel. Based on the novel zero-knowledge proof system [94], it constructed a special-function smart contract to guarantee privacy properties that are able to resist curious users included in the payment path from the payer to the beneficiary. However, this work is inefficient since it needs to exchange a large amount of data between the users in the payment path, which degrades its performance.

In Table 4, we compare all efforts of privacy-preserving off-chain channels based on criteria listed in section 3 in order to give a comprehensive overview.

4.3. Smart contract privacy

As a decentralized computer program that runs upon the blockchain, the smart contract extends the function of blockchain beyond cryptocurrency. Because the entire process of contract execution is transparent to all and will be permanently recorded on the blockchain, smart contracts based on the blockchain confront serious privacy risks.

To address the privacy issues in the blockchain-based smart contract, Kosba et al. proposed the first privacy-preserving smart contract platform called Hawk [95]. It provides an easy way for developers to build a private smart contract without using any obfuscation techniques or code encryption. Hawk divides the smart contract into two portions: the private part and the public part. The private part is responsible for the secret data or functions involved in a contract, and the public part is responsible for the public codes that are transparent to external entities. The main protocol includes a particular party, named the manager, built with Intel Software Guard Extensions (SGX) to facilitate the execution of private part. Due to the confidentiality of SGX's data, the manager can obtain private information and the entire sequence of transaction actions during the execution of the contract, but they will not be disclosed. If the manager suspends the protocol, it will be automatically financially penalized. Hawk leverages the zk-SNAKRs to ensure the correctness of funds' transfer and contract execution, which also results in a relatively high computational overhead. Besides Hawk requires the user to use a coin and cannot be deployed directly on most blockchain systems because of their low efficiency.

Cryptographic solutions usually cause severe performance degradation. Therefore, few of them can be applied directly in permissionless blockchain due to its limited capacity. In order to address this issue, some works employ secure hardware to protect privacy. ShadowEth [96] employs Trusted Execution Environment (TEE) for privacy-preserving smart contracts on Ethereum. ShadowEth allows users to create bounty contracts that are executed within TEE and store all metadata in a TEE-based off-chain storage system called TEE-DS. Ekiden [97] also utilizes secure hardware but further improves the efficiency, and hence has high performance. Ekiden is the first privacy-preserving smart contract system with a throughput of more than 1000 transactions per second. By combing the trusted hardware and blockchain, Ekiden can be deployed to different blockchain systems (permissionless or permissioned blockchain). Since it operates computing nodes in off-chain TEEs, it avoids the long latency and high computational burden of the on-chain execution. The cryptographic verification process in Hawk is replaced by validating remote attestations to provide verifiable computation. The two schemes improve efficiency while solving privacy issues. However, privacy preservation depends on the security of trusted hardware, and once the trusted hardware is compromised, these schemes will become ineffective.

To overcome the weaknesses of the above schemes, Arbitrum [98] relied on the Virtual Machine (VM) to implement the contract's functionality and simultaneously protect privacy. It allows a user to implement the private smart contract as a VM that encodes contract rules. Arbitrum includes an incentive mechanism that encourages users to agree on the off-chain of VM behavior. As a result, the Arbitrum miners confirm the agreement by only verifying digital signatures. Unlike Ethereum, verifiers in Arbitrum can efficiently verify transactions without revealing any internal state of a VM and settle disputes about contract behaviors with only examining one instruction for every execution of the contract. Therefore, it improves dramatically in privacy and scalability. However, its incentive mechanism is effective only if most managers are rational.

In Table 5, we compare all efforts of privacy-preserving smart contracts discussed in this part.

5. Open issues and future directions

Based on the analysis and comparison results, we summarize the unsolved issues in the privacy preservation of permissionless blockchain. Besides, we also propose a series of future research directions.

5.1. Open research issues

According to the above analysis and comparison in Section 4, we find several unsolved issues in privacy preservation in permissionless blockchains.

First, in terms of performance, although many privacy solutions try to improve efficiency, the computation overhead is still quite high for a permissionless blockchain system. Many cryptographic tools, such as zero-knowledge proof, have problems such as large transaction size and long transaction processing time, which are not suitable for large-scale applications. Besides, it is not efficient to apply them to instant applications due to their long delay.

Second, existing works usually ignore the necessity of accountability and conditional traceability. Privacy protection grants user's freedom to make payments without being recognized by non-participants, which makes it possible to employ the blockchain to conduct crimes, such as drug/weapon trading. Therefore, it is necessary to disclose the real identity of malicious nodes in some cases. However, unlike centralized architecture, the permissionless blockchain lacks a powerful and trusted party to offer privacy insurance and meanwhile works as an arbitral authority. The decentralized architecture provides attackers with more chances and methods to conduct misbehaviors, and it becomes more challenging to solve the conflict between privacy and accountability. Nevertheless, existing works seldom consider this issue.

Third, as the most groundbreaking technology involved with the blockchain, the smart contract requires privacy preservation in many scenarios. Existing schemes either need to assume the security and trust of SGX or utilize heavy cryptographic tools with high computation/storage overhead. For SGX, its security cannot be fully ensured as claimed

Table 4
Comparison of privacy-preserving off-chain channel.

Proposals	Privacy Protection	Compatibility	Functionality	Channel Direction	Disadvantages
Heilman's work [63]	External anonymity	Compatible with Bitcoin	Payment hub	Unidirectional	Need to assume the intermediary will not violate rules
Bolt [91]	Internal/external anonymity	Not compatible with Bitcoin	Payment hub	Unidirectional/bidirectional	Privacy relies on underlying cryptocurrency
Tumblebit [64]	Internal/external anonymity	Compatible with Bitcoin	Payment hub	Unidirectional	Does not support arbitrary denomination and payment value hiding
Giulio's work [93]	Internal/external anonymity	Compatible with Bitcoin	Payment channel network	Unidirectional/bidirectional	Need to exchange large amount of data

Table 5
Comparison of privacy-preserving smart contract.

Proposals	Privacy Protection	Compatibility	Techniques Based	Parties executing smart contract	Disadvantages
Hawk [95]	Hiding transaction amount, identities of participants and contract input, state from non-participants	Not compatible with Ethereum	SGX ZK-SNARKs	Single SGX-enabled manager	Requires trusting the security of Intel SGX and issuer of the attestation keys (e.g., Intel), the supporting range of contract types is limited, and the size of proof limits its performance
ShadowEth [96]	Hiding contract input, state from non-participants	Compatible with Ethereum	SGX	Multiple SGX-enabled worker nodes	Requires trusting the security of Intel SGX and issuer of the attestation keys (e.g., Intel)
Ekiden [97]	Hiding contract input, state from non-participants	Not compatible with Ethereum	SGX	Multiple SGX-enabled compute nodes	Requires trusting the security of Intel SGX and issuer of the attestation keys (e.g., Intel)
Arbitrum [98]	Hiding contract state from non-participants	Not compatible with Ethereum	VM	Multiple nodes running VM	Require assumption that at least one manager is honest and the rest of the managers are rational

since there are already several works that effectively obtain the secret protected by it [99,100]. Besides, the security of TEE's operation relies on the integrity of Intel, which introduces the risk of a single point of failure and is not suitable to the decentralization property of Blockchain. While as analyzed, cryptographic tools suffer from high computation or storage overhead and are not suitable for many applications. Therefore, the protection of smart contract privacy remains as an open issue.

5.2. Future research directions

In this subsection, we suggest some future research directions based on the open research issues.

5.2.1. Privacy preservation with high efficiency

Privacy Preservation with high efficiency remains an unsolved issue, which significantly constrains the practical deployment of the blockchain. The limited computation capacity of blockchain makes it unable to conduct computationally expensive cryptographic operations. Besides, the schemes should also reduce the number of transactions recorded in blockchain to reduce expenses. Therefore, efficient privacy preservation schemes are highly expected. However, the decentralization, transparency, and inefficiency make it challenging to achieve efficient privacy preservation, which should be further explored in the future.

5.2.2. Privacy preservation with accountability and decentralization

Accountability is rarely explored by existing works. However, accountability is required in many application scenarios. It is necessary to disclose the identities of malicious users in the case that blockchain becomes a platform for crimes. Besides, some scenarios require trust evaluation on users or auditing on data, which is not supported by most privacy preservation. Obviously, accountability is in conflict with privacy, and we should carefully weigh accountability and privacy when designing a scheme with privacy preservation and accountability. Besides, the blockchain lacks a powerful and trusted centralized party for privacy insurance. Implementing accountability with a centralized party or trusted hardware is easy, but introduces the risk of a single point of failure. Considering the necessity of accountability and its challenges, decentralized privacy preservation with accountability would be a significant future research direction.

5.2.3. Privacy preservation for smart contract privacy

The smart contract is the key technology for building various blockchain-based applications. It requires miners to verify the correctness of execution results of a contract, which makes the preservation of contract privacy more complicated. The contract itself and the data generated during the contract execution should be kept inviable to all except the contract creators. Current schemes employ verifiable computing, SMC, or trusted hardware. As analyzed, they cannot entirely fit with blockchain since they are not efficient enough or rely on centralized parties. It is necessary to preserve contract privacy in a decentralized and efficient way to make it practical to be deployed in the real world.

6. Conclusions

The blockchain has been widely used in various fields because of its decentralization, data immutability, and trustworthiness. However, transparency and decentralization make it difficult to protect user privacy effectively, which makes privacy preservation in blockchain an important research topic, especially for permissionless blockchain. In this paper, we first analyzed the privacy issues in the permissionless blockchain and summarized the potential threats to privacy. We then proposed a series of evaluation criteria, with which we discussed the advantages and disadvantages of the state-of-the-art work. Based on the analysis and comparison of the results, we found several open issues and proposed a series of future research directions, which are helpful for the research on practical blockchain systems with privacy preservation.

Declaration of competing interest

We claim that we have no conflict of interest with other researchers with regard to the paper.

Acknowledgements

The work is supported in part by the National Natural Science Foundation of China under Grants 61672410 and 61802293, the Academy of Finland under Grants 308087, 314203 and 335262, the Key Lab of Information Network Security, Ministry of Public Security under grant No. C18614, the open grant of the Tactical Data Link Lab of the 20th Research Institute of China Electronics Technology Group Corporation, P.R. China under grant CLDL-20182119, the National Postdoctoral Program for Innovative Talents under grant BX20180238, the Project funded by China Postdoctoral Science Foundation under grant 2018M633461, the Shaanxi Innovation Team project under grant 2018TD-007, and the 111 project under grant B16037.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.dcan.2020.05.008.

References

[1] M. Crosby, P. Pattanayak, S. Verma, V. Kalyanaraman, et al., Blockchain technology: beyond bitcoin, Appl. Innov. 2 (6-10) (2016) 71.
[2] S. Huh, S. Cho, S. Kim, Managing iot devices using blockchain platform, in: 2017 19th International Conference on Advanced Communication Technology, ICACT), 2017, pp. 464–467. IEEE.
[3] A. Dorri, S.S. Kanhere, R. Jurdak, Towards an optimized blockchain for iot, in: Proceedings of the Second International Conference on Internet-Of-Things Design and Implementation, ACM, 2017, pp. 173–178.

[4] M. Samaniego, R. Deters, Blockchain as a service for iot, in: IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data, SmartData), 2016, pp. 433–436. IEEE, 2016.
[5] P.K. Sharma, M.-Y. Chen, J.H. Park, A software defined fog node based distributed blockchain cloud architecture for iot, IEEE Access 6 (2017) 115–124.
[6] Z. Lu, Q. Wang, G. Qu, Z. Liu, Bars: a blockchain-based anonymous reputation system for trust management in vanets, in: 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/12th IEEE International Conference on Big Data Science and Engineering, TrustCom/BigDataSE), 2018, pp. 98–103. IEEE, 2018.
[7] Z. Lu, W. Liu, Q. Wang, G. Qu, Z. Liu, A privacy-preserving trust model based on blockchain for vanets, IEEE Access 6 (2018) 45655–45664.
[8] K. Biswas, V. Muthukkumarasamy, Securing smart cities using blockchain technology, in: IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City, IEEE 2nd international conference on data science and systems (HPCC/SmartCity/DSS), 2016, pp. 1392–1393. IEEE, 2016.
[9] P.K. Sharma, S.Y. Moon, J.H. Park, Block-vn, A distributed blockchain based vehicular network architecture in smart city, JIPS 13 (1) (2017) 184–195.
[10] G. Liu, H. Dong, Z. Yan, X. Zhou, S. Shimizu, B4sdc: A Blockchain System for Security Data Collection in Manets, IEEE Transactions on Big Data.
[11] W. Feng, Y. Li, X. Yang, Z. Yan, L. Chen, Blockchain Based Data Transmission Control for Tactical Datalink, Digital Communications and Networks.
[12] K.K.R. Choo, Z. Yan, W.Z. Meng, Blockchain in Industrial IoT Applications Security and Privacy Advances, Challenges and Opportunities, IEEE Trans. Ind. Inf. 16 (6) (2020) 4119–4121.
[13] Z. Yan, X. Huang, A. V. Vasilakos, L. T. Yang, Special issue on blockchain and decentralization for internet of things, Future Generat. Comput. Syst..
[14] Z. Zheng, S. Xie, H. Dai, X. Chen, H. Wang, An overview of blockchain technology: architecture, consensus, and future trends, in: IEEE International Congress on Big Data (BigData Congress), 2017, pp. 557–564. IEEE, 2017.
[15] D. Ron, A. Shamir, Quantitative analysis of the full bitcoin transaction graph, in: International Conference on Financial Cryptography and Data Security, Springer, 2013, pp. 6–24.
[16] I.-C. Lin, T.-C. Liao, A survey of blockchain security issues and challenges, IJ Netw. Sec. 19 (5) (2017) 653–659.
[17] A. Urquhart, The inefficiency of bitcoin, Econ. Lett. 148 (2016) 80–82.
[18] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, N. Zeldovich, Algorand: scaling byzantine agreements for cryptocurrencies, in: Proceedings of the 26th Symposium on Operating Systems Principles, ACM, 2017, pp. 51–68.
[19] I. Eyal, A.E. Gencer, E.G. Sirer, R. Van Renesse, Bitcoin-ng: a scalable blockchain protocol, in: 13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16), 2016, pp. 45–59.
[20] Q. Feng, D. He, S. Zeadally, M.K. Khan, N. Kumar, A survey on privacy protection in blockchain system, J. Netw. Comput. Appl. 126 (2019) 45–58.
[21] D. Yang, J. Gavigan, Z. Wilcox-O'Hearn, Survey of confidentiality and privacy preserving technologies for blockchains, R3 Res. 1 (2016) 1–10.
[22] M. Conti, E.S. Kumar, C. Lal, S. Ruj, A survey on security and privacy issues of bitcoin, IEEE Commun. Surv. Tutor. 20 (4) (2018) 3416–3452.
[23] R. Zhang, R. Xue, L. Liu, Security and privacy on blockchain, ACM Comput. Surv. 52 (3) (2019) 1–34.
[24] X. Li, P. Jiang, T. Chen, X. Luo, Q. Wen, A survey on the security of blockchain systems, Future Generat. Comput. Syst. 12 (3) (2017) 1–33.
[25] R.C. Merkle, A digital signature based on a conventional encryption function, in: Conference on the Theory and Application of Cryptographic Techniques, Springer, 1987, pp. 369–378.
[26] L. Bach, B. Mihaljevic, M. Zagar, Comparative analysis of blockchain consensus algorithms, in: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), IEEE, 2018, pp. 1545–1550.
[27] Y. Xiao, N. Zhang, W. Lou, Y. T. Hou, Modeling the Impact of Network Connectivity on Consensus Security of Proof-Of-Work Blockchain, arXiv preprint arXiv:2002.08912.
[28] S. Nakamoto, et al., Bitcoin: a peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, 2019.
[29] G. Wood, et al., Ethereum: a secure decentralised generalised transaction ledger, Ether. Project Yellow Pap. 151 (2014) 1–32, 2014.
[30] K. Christidis, M. Devetsikiotis, Blockchains and smart contracts for the internet of things, IEEE Access. 4 (2016) 2292–2303.
[31] M. Swan, Blockchain: Blueprint for a New Economy, O'Reilly Media, Inc., 2015.
[32] B. Chen, Z. Tan, W. Fang, Blockchain-based implementation for financial product management, in: 2018 28th International Telecommunication Networks and Applications Conference (ITNAC), IEEE, 2018, pp. 1–3.
[33] Q.K. Nguyen, Blockchain-a financial technology for future sustainable development, in: 3rd International Conference on Green Technology and Sustainable Development (GTSD), IEEE, 2016, pp. 51–54, 2016.
[34] R. Lewis, J. McPartland, R. Ranjan, Blockchain and financial market innovation, Econ. Perspect. 41 (7) (2017) 1–17.
[35] R. Hanifatunnisa, B. Rahardjo, Blockchain based e-voting recording system design, in: 11th International Conference on Telecommunication Systems Services and Applications (TSSA), IEEE, 2017, pp. 1–6, 2017.
[36] R. Osgood, The Future of Democracy: Blockchain Voting, Information Security, COMP116, 2016, pp. 1–21.
[37] N. Kshetri, J. Voas, Blockchain-enabled e-voting, IEEE Software 35 (4) (2018) 95–99.
[38] A. Sutton, R. Samavi, Blockchain enabled privacy audit logs, in: International Semantic Web Conference, Springer, 2017, pp. 645–660.
[39] K. Fan, S. Sun, Z. Yan, Q. Pan, H. Li, Y. Yang, A blockchain-based clock synchronization scheme in, Future Generat. Comput. Syst. 101 (2019) 524–533.
[40] K. Fan, S. Wang, Y. Ren, K. Yang, Z. Yan, H. Li, Y. Yang, Blockchain-based secure time protection scheme in iot, IEEE Int. Things J. 6 (3) (2018) 4671–4679.
[41] K. Fan, Y. Ren, Z. Yan, S. Wang, H. Li, Y. Yang, Secure time synchronization scheme in iot based on blockchain, in: IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data, IEEE, 2018, pp. 1063–1068 (SmartData),2018.
[42] W. Feng, Z. Yan, Mcs-chain: decentralized and trustworthy mobile crowdsourcing based on blockchain, Future Generat. Comput. Syst. 95 (2019) 649–666.
[43] Y. Zhang, R.H. Deng, J. Shu, K. Yang, D. Zheng, Tkse: trustworthy keyword search over encrypted data with two-side verifiability via blockchain, IEEE Access 6 (2018) 31077–31087.
[44] C. Cai, X. Yuan, C. Wang, Hardening distributed and encrypted keyword search via blockchain, in: IEEE Symposium on Privacy-Aware Computing, PAC), 2017, pp. 119–128. IEEE, 2017.
[45] C. Cai, X. Yuan, C. Wang, Towards trustworthy and private keyword search in encrypted decentralized storage, in: IEEE International Conference on Communications (ICC), IEEE, 2017, pp. 1–7, 2017.
[46] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, L. Njilla, 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, pp. 468–477.
[47] H. Shafagh, L. Burkhalter, A. Hithnawi, S. Duquennoy, Towards blockchain-based auditable storage and sharing of iot data, in: Proceedings of the 2017 on Cloud Computing Security Workshop, ACM, 2017, pp. 45–50.
[48] M. Ali, J. Nelson, R. Shea, M.J. Freedman, Blockstack: a global naming and storage system secured by blockchains, in: {USENIX} Annual Technical Conference ({USENIX}{ATC} 16), 2016, 2016, pp. 181–194.
[49] M. El-Hindi, C. Binnig, A. Arasu, D. Kossmann, R. Ramamurthy, Blockchaindb: a shared database on blockchains, Proc. VLDB Endow. 12 (11) (2019) 1597–1609.
[50] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, Y. Zhang, Consortium blockchain for secure energy trading in industrial internet of things, Proc. Trans. Indus. Inf. 14 (8) (2017) 3690–3700.
[51] W. Feng, Y. Li, X. Yang, Z. Yan, L. Chen, Blockchain based data transmission control for tactical data link, in: International Conference on Smart City and Informatization, Springer, 2019, pp. 583–595.
[52] Z. Yan, L. Peng, Trust evaluation based on blockchain in pervasive social networking, IEEE Block. News. (2018) 1–4.
[53] F. Reid, M. Harrigan, An analysis of anonymity in the bitcoin system, in: Security and Privacy in Social Networks, Springer, 2013, pp. 197–223.
[54] S. Goldfeder, H. Kalodner, D. Reisman, A. Narayanan, When the cookie meets the blockchain: privacy risks of web payments via cryptocurrencies, Proc. Privacy Enhanc. Technol. (4) (2018) 179–199, 2018.
[55] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G.M. Voelker, S. Savage, A fistful of bitcoins: characterizing payments among men with no names, in: Proceedings of the 2013 Conference on Internet Measurement Conference, ACM, 2013, pp. 127–140.
[56] P. Koshy, D. Koshy, P. McDaniel, An analysis of anonymity in bitcoin using p2p network traffic, in: International Conference on Financial Cryptography and Data Security, Springer, 2014, pp. 469–485.
[57] A. Pfitzmann, M. Hansen, A terminology for talking about privacy by data minimization: anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. http://www.maroki.de/pub/dphistory/2010_Anon_Terminology_v0.34.pdf, 2010.
[58] D. Chaum, Untraceable electronic mail, return addresses and digital pseudonyms, in: Secure Electronic Voting, Springer, 2003, pp. 211–219.
[59] J. Bonneau, A. Narayanan, A. Miller, J. Clark, J.A. Kroll, E.W. Felten, Mixcoin: anonymity for bitcoin with accountable mixes, in: International Conference on Financial Cryptography and Data Security, Springer, 2014, pp. 486–504.
[60] L. Valenta, B. Rowan, Blindcoin: blinded, accountable mixes for bitcoin, in: International Conference on Financial Cryptography and Data Security, Springer, 2015, pp. 112–126.
[61] E. Duffield, D. Diaz, Dash: a privacycentric cryptocurrency. https://github.com/dashpay/dash/wiki/Whitepaper, 2015.
[62] G. Maxwell, Coinswap: transaction graph disjoint trustless trading. https://bitcointalk.org/index.php?topic=321228.0, 2013.
[63] E. Heilman, F. Baldimtsi, S. Goldberg, Blindly signed contracts: anonymous on-blockchain and off-blockchain bitcoin transactions, in: International Conference on Financial Cryptography and Data Security, Springer, 2016, pp. 43–60.
[64] E. Heilman, L. Alshenibr, F. Baldimtsi, A. Scafuro, S. Goldberg, Tumblebit: an untrusted bitcoin-compatible anonymous payment hub, in: Network and Distributed System Security Symposium, 2017.
[65] G. Maxwell, Coinjoin: bitcoin privacy for the real world. https://bitcointalk.org/index.php?topic=279249.0, 2013.
[66] T. Ruffing, P. Moreno-Sanchez, A. Kate, Coinshuffle: practical decentralized coin mixing for bitcoin, in: European Symposium on Research in Computer Security, Springer, 2014, pp. 345–364.
[67] H. Corrigan-Gibbs, B. Ford, Dissent: accountable anonymous group messaging, in: Proceedings of the 17th ACM Conference on Computer and Communications Security, ACM, 2010, pp. 340–350.

[68] J.H. Ziegeldorf, F. Grossmann, M. Henze, N. Inden, K. Wehrle, Coinparty: secure multi-party mixing of bitcoins, in: Proceedings of the 5th ACM Conference on Data and Application Security and Privacy, ACM, 2015, pp. 75–86.

[69] D. Chaum, The dining cryptographers problem: Unconditional sender and recipient untraceability, J. Cryptol. 1 (1) (1988) 65–75.

[70] P. Golle, A. Juels, Dining cryptographers revisited, in: International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2004, pp. 456–473.

[71] T. Ruffing, P. Moreno-Sanchez, A. Kate, P2p mixing and unlinkable bitcoin transactions, in: NDSS, 2017, pp. 511–532.

[72] R.L. Rivest, A. Shamir, Y. Tauman, How to leak a secret, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2001, pp. 552–565.

[73] E. Bresson, J. Stern, M. Szydlo, Threshold ring signatures and applications to ad-hoc groups, in: Annual International Cryptology Conference, Springer, 2002, pp. 465–480.

[74] N. Van Saberhagen, Cryptonote v 2.0. https://static.coinpaprika.com/storage/cdn/whitepapers/1611.pdf, 2013.

[75] E. Fujisaki, K. Suzuki, Traceable ring signature, in: International Workshop on Public Key Cryptography, Springer, 2007, pp. 181–200.

[76] S. Noether, Ring signature confidential transactions for monero, IACR Cryptol. ePrint Archiv. (2015) 1098, 2015.

[77] G. Maxwell, Confidential Transactions, (Accessed 09/05/2016).

[78] M. Möser, K. Soska, E. Heilman, K. Lee, H. Heffan, S. Srivastava, K. Hogan, J. Hennessey, A. Miller, A. Narayanan, et al., An empirical analysis of traceability in the monero blockchain, Proc. Privacy Enhanc. Technol. (3) (2018) 143–163, 2018.

[79] S.-F. Sun, M.H. Au, J.K. Liu, T.H. Yuen, Ringct 2.0: a compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency monero, in: European Symposium on Research in Computer Security, Springer, 2017, pp. 456–474.

[80] T.H. Yuen, S.-f. Sun, J.K. Liu, M.H. Au, M.F. Esgin, Q. Zhang, D. Gu, Ringct 3.0 for blockchain confidential transaction: shorter size and stronger security, Tech. rep., Cryptology ePrint Archive (2019). Report 2019/508.(2019). **Error! Hyperlink reference not valid.**.

[81] S. Goldwasser, S. Micali, C. Rackoff, The knowledge complexity of interactive proof systems, SIAM J. Comput. 18 (1) (1989) 186–208.

[82] I. Miers, C. Garman, M. Green, A.D. Rubin, Zerocoin: anonymous distributed e-cash from bitcoin, in: IEEE Symposium on Security and Privacy, IEEE, 2013, pp. 397–411, 2013.

[83] R. Cramer, I. Damgård, B. Schoenmakers, Proofs of partial knowledge and simplified design of witness hiding protocols, in: Annual International Cryptology Conference, Springer, 1994, pp. 174–187.

[84] E. Androulaki, G.O. Karame, Hiding transaction amounts and balances in bitcoin, in: International Conference on Trust and Trustworthy Computing, Springer, 2014, pp. 161–178.

[85] E.B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, M. Virza, Zerocash: decentralized anonymous payments from bitcoin, in: IEEE Symposium on Security and Privacy, IEEE, 2014, pp. 459–474, 2014.

[86] E. Ben-Sasson, A. Chiesa, D. Genkin, E. Tromer, M. Virza, Snarks for c: verifying program executions succinctly and in zero knowledge, in: Annual Cryptology Conference, Springer, 2013, pp. 90–108.

[87] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, G. Maxwell, Bulletproofs: short proofs for confidential transactions and more, in: IEEE Symposium on Security and Privacy (SP), IEEE, 2018, pp. 315–334, 2018.

[88] A. Jivanyan, Lelantus: towards confidentiality and anonymity of blockchain transactions from standard assumptions, IACR Cryptol. ePrint Archiv. (2019) 373, 2019.

[89] B. Bünz, S. Agrawal, M. Zamani, D. Boneh, Zether: towards privacy in a smart contract world, IACR Cryptol. ePrint Archiv. (2019) 191, 2019.

[90] J. Spilman, Anti dos for tx replacement. https://lists.linuxfoundation.org/pipermail/bitcoin-dev/2013-April/002417.html, 2013.

[91] M. Green, I. Miers, Bolt: anonymous payment channels for decentralized currencies, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, ACM, 2017, pp. 473–489.

[92] J. Camenisch, S. Hohenberger, A. Lysyanskaya, Compact e-cash, in: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2005, pp. 302–321.

[93] G. Malavolta, P. Moreno-Sanchez, A. Kate, M. Maffei, S. Ravi, Concurrency and privacy with payment-channel networks, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, ACM, 2017, pp. 455–471.

[94] I. Giacomelli, J. Madsen, C. Orlandi, Zkboo: faster zero-knowledge for boolean circuits, in: 25th {USENIX} Security Symposium ({USENIX} Security 16), 2016, pp. 1069–1083.

[95] A. Kosba, A. Miller, E. Shi, Z. Wen, C. Papamanthou, Hawk, The blockchain model of cryptography and privacy-preserving smart contracts, in: IEEE Symposium on Security and Privacy, SP), 2016, pp. 839–858. IEEE, 2016.

[96] R. Yuan, Y.-B. Xia, H.-B. Chen, B.-Y. Zang, J. Xie, Shadoweth: private smart contract on public blockchain, J. Comput. Sci. Technol. 33 (3) (2018) 542–556.

[97] R. Cheng, F. Zhang, J. Kos, W. He, N. Hynes, N. Johnson, A. Juels, A. Miller, D. Song, Ekiden: a platform for confidentiality-preserving, trustworthy, and performant smart contracts, in: IEEE European Symposium on Security and Privacy (EuroS&P), IEEE, 2019, pp. 185–200, 2019.

[98] H. Kalodner, S. Goldfeder, X. Chen, S.M. Weinberg, E.W. Felten, Arbitrum: scalable, private smart contracts, in: 27th {USENIX} Security Symposium ({USENIX} Security 18), 2018, pp. 1353–1370.

[99] M. Lipp, M. Schwarz, D. Gruss, T. Prescher, W. Haas, A. Fogh, J. Horn, S. Mangard, P. Kocher, D. Genkin, et al., Meltdown: reading kernel memory from user space, in: 27th {USENIX} Security Symposium ({USENIX} Security 18), 2018, pp. 973–990.

[100] P. Kocher, J. Horn, A. Fogh, D. Genkin, D. Gruss, W. Haas, M. Hamburg, M. Lipp, S. Mangard, T. Prescher, et al., Spectre attacks: exploiting speculative execution, in: IEEE Symposium on Security and Privacy (SP), IEEE, 2019, pp. 1–19, 2019.