A Survey on Anonymity and Privacy in Bitcoin-Like Digital Cash Systems

Merve Can Kus Khalilov and Albert Levi, *Member, IEEE*

***Abstract*—Bitcoin is the most widely known distributed, peer-to-peer payment network without existence of a central authority. In Bitcoin, users do not use real names; instead, pseudonyms are used. Managing and verifying transactions and issuing of bitcoins are performed collectively by peers in the network. Since pseudonyms are used without providing any identity, it is perceived that Bitcoin provides anonymity. However, it is one of the most transparent payment networks since all transactions are publicly announced. Blockchain, which is the public ledger of Bitcoin, includes all transactions to prevent double-spending and to provide integrity. By using data in the blockchain, flow of bitcoins between transactions can be observed and activities of the users can be traced. When the implications obtained from the blockchain are combined with external data, identity and profile of a user can be revealed. This possibility has undesirable effects such as spending history of a user becomes accessible to other people, or cash flow of a merchant becomes exposed to competitors. There are several proposals as extensions or alternatives to Bitcoin, which improve anonymity and privacy. This survey presents an overview and detailed investigation of anonymity and privacy in Bitcoin-like digital cash systems. We examine the studies in the literature/Web in two major categories: 1) analyses of anonymity and privacy in Bitcoin and 2) extensions and alternatives to Bitcoin, which improve anonymity and privacy. We list and describe methods and outcomes for both categories and group studies according to these methods and outcomes. We also present relationships between outcomes of analyses and the improvement methods. We compare performances of the methods and show relationships between the proposals. Moreover, we present guidelines for designing an anonymity/privacy improvement and discuss future research directions.**

***Index Terms*—Anonymity and privacy, Bitcoin, blockchain, cryptocurrency, digital cash, peer-to-peer computing.**

I. Introduction

PAYMENTS, money transfers and commerce are widely preferred to be made on the Internet for a long while, especially in the last decade, as with many other things in this digital age. This preference comes from the speed, which is provided by the digitalization and needed in busy daily life, as well as increasing global connectivity with the rise of digital businesses and social networks. Commerce on the Internet is done assured that financial institutions and banks serve as trusted authorities. This model can be called trust-based; buyers and merchants may not trust each other, however, they trust well-known banks and banks act as trust entities managing transactions and keeping records. However, there are some disadvantages with this trust-based model [1]. First, financial institutions act as mediators between merchants and buyers, and there exists a cost for mediation. This limits the minimum practical transaction size. Second, there is a possibility of reversal of transactions. Transactions can be reversed by banks if there is a dispute between the trading parties, e.g., the buyer transfers the money, but the seller does not send goods or provide services to the buyer. However, this possibility of reversal compels that merchants get information about their customers. On the contrary, merchants do not have to get extra information about their customers like billing address, name, etc. when transactions are irreversible. In addition, irreversible transactions protect merchants from *chargeback fraud*, i.e., a dishonest buyer says that he did not make the purchase [2]. If dishonest merchants are considered, using escrow services may be a method for protecting buyers in the case of irreversible transactions. Finally, especially international transactions in the regular banking, e.g., money transfers, are slow due to procedural delays.

An electronic payment system, which is not based on trust, can be realized using cryptographic mechanisms. Digital cash concept, which utilizes cryptography, was first introduced by Chaum [3] in 1982 and evolved from trust-based model to decentralized networks in the subsequent decades. In a decentralized system, where there is not any trusted authority, parties, also called peers, transact directly with each other forming a Peer-to-Peer (P2P) network. This kind of digital cash systems, which use virtual assets and utilize cryptography, are also called cryptocurrency. Bitcoin [1], which is introduced by Satoshi Nakamoto[1] in 2008, is the first

Manuscript received July 3, 2017; revised December 23, 2017 and February 8, 2018; accepted March 18, 2018. Date of publication March 26, 2018; date of current version August 21, 2018. *(Corresponding author: Merve Can Kus Khalilov.)*

M. C. Kus Khalilov is with the Research and Development Center, Kuveyt Turk Participation Bank, 41420 Çayirova, Turkey, and also with the Computer Science and Engineering Department, Faculty of Engineering and Natural Sciences, Sabanci University, 34956 Istanbul, Turkey (e-mail: mervecank@sabanciuniv.edu).

A. Levi is with the Computer Science and Engineering Department, Faculty of Engineering and Natural Sciences, Sabanci University, 34956 Istanbul, Turkey (e-mail: levi@sabanciuniv.edu).

Digital Object Identifier 10.1109/COMST.2018.2818623

[1]The real identity of Satoshi Nakamoto has remained obscure since 2008. It is uncertain that the name is used by a person or a group of people, as well. In 2010, Nakamoto handed over control of the source code repository and several related domains to Gavin Andresen and various prominent members of the Bitcoin community and stopped his involvement in the project [4]. On his P2P Foundation profile as of 2012, Nakamoto claimed to be a 37-year-old male who lived in Japan; however, his use of perfect British English raised claims that he was of Commonwealth origin. There is still doubt about the real identity of Satoshi Nakamoto. Some individuals were thought to be Nakamoto, but these people denied being him. Finally and recently, on 2nd May 2016, Australian entrepreneur Craig Wright publicly claimed to be Nakamoto, after some findings pointed at him [5]. He provided some proof, even though, this announcement was met with strong skepticism. This led Wright to retreat; he deleted his blog, that included his claims, and replaced it with an apology, writing that he did not "have the courage" to continue to try to prove his case [6], [7].

1553-877X © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

decentralized digital currency. Although a huge number of alternative proposals emerged following the ideas of Bitcoin, Bitcoin is still the most widely accepted and used one. There can be found a total number of 1372 cryptocurrencies listed in CryptoCurrency Market Capitalizations Web site [8] with $606 billion total market cap, where the number of circulating bitcoins exceeded 16.7 million, and the total Bitcoin market cap exceeded $268 billion [9], resulting Bitcoin dominance over 44% as of December 2017.

Bitcoin is defined as "an innovative payment network and a new kind of money" [10]. Essentially, it is an open source P2P money. In Bitcoin, transactions are recorded in a publicly distributed ledger, which is called *blockchain*. The base unit of account is *bitcoin*, and the lowest-valued unit is *satoshi*. There is no central authority or a bank, managing and verifying transactions. These operations and issuing of bitcoins are performed collectively by the network, which consists of communicating nodes (peers) running Bitcoin software. Blockchain structure provides a single and shared history for all users, which also provides integrity. Issuing bitcoins is achieved through *mining* process. Mining is the activity of adding transaction records to the blockchain. Users spend their computing power to verify and record payments; in return, they earn bitcoins, which are created as the result of this payment processing work as a reward. Bitcoins can also be exchanged for other currencies or used for buying products and services. Transactions are computationally impractical to reverse, and these non-reversible transactions protect sellers from fraud.

Users are not required to provide real names to use Bitcoin. Instead, pseudonyms are used, so it is explicitly seen that some entities transact with each other, but the real identities stay hidden like in stock exchange operations. However, since all transactions are publicly available, activities of the users can be tracked and linked. Therefore, profile of the users can be extracted, and the user identities can be revealed by linking one of the transactions to off-network information, as clearly shown in the previous studies that analyze anonymity and privacy in Bitcoin. Therefore, users cannot stay completely anonymous, and user privacy is not provided since amount values, sender and receiver user addresses are explicitly visible in the blockchain. This shortcoming, which introduces the possibility of tracing, results in, for example, spending history of a user becoming available and accessible to all other people, or cash flow of merchants becoming exposed to their competitors. All this transparency also causes bitcoins not to be equal, i.e., each bitcoin is associated with different transactions. This can lead to, for instance, a user not accepting some bitcoins for their tainted history, say, if they are associated with illegal activity. This deficit that is by design of Bitcoin caused the emergence of several proposals with improved anonymity and privacy.

Although Bitcoin has emerged to be used in the financial sector, its blockchain structure and peer to peer network attracted the academic community, as well. This attraction resulted in numerous studies on Bitcoin taking place in the literature.

In this survey, the studies, related to anonymity and privacy issues in Bitcoin, are examined thoroughly. We covered the studies that we encountered on the Web up to April 2017. To the best of our knowledge, there exist six previous studies that survey the literature related to Bitcoin anonymity and privacy. A very brief survey by Schaffner [11], which is included in resources of a lecture, mentioned previous work on Bitcoin security and anonymity; however, the content and the number of studies is very limited. ShenTu and Yu [12] compiled research on anonymization and deanonymization in the Bitcoin system. They grouped deanonymization methods, like linking Bitcoin addresses expected to belong to the same user by blockchain analysis or network analysis, and countermeasure methods. They mentioned studies including these methods; however, there still are missing studies in this work, which we covered in our survey. Herrera-Joancomartí [13] examined studies on Bitcoin anonymity and grouped them into three classes; (*i*) blockchain analysis, (*ii*) traffic analysis and (*iii*) mixing techniques for anonymity. However, the number of the included studies is very small, and there is no further classification, or there are not any proposed properties for analyzing the studies. A very detailed examination of Bitcoin and similar cryptocurrencies from many perspectives were provided by Bonneau *et al.* [14]. However, the discussion has a broader perspective compared to this study. Stability, client-side security, and alternative modification methods to the protocol are some included topics. Although anonymity and privacy issues are also included in a separate chapter analyzing the studies related to deanonymization and proposals for improving anonymity, it is kept brief, and a very limited number of studies are examined. In addition, very few studies are compared according to five properties. Narayanan *et al.* published a book [15] providing introductory information on Bitcoin and cryptocurrency technologies that includes a section on Bitcoin and anonymity. An extensive technical survey by Tschorsch and Scheuermann [16] provided a more in-depth overview by including more studies. Nevertheless, it is not focused on anonymity and privacy, as well. Issues on security and network are discussed, and detailed investigation on Proof-of-X (PoX) schemes are done. With regard to anonymity and privacy, studies investigating deanonymization, blockchain analysis and enabling privacy are surveyed in more detail as compared to [14]. However, related literature has not been classified as detailed as the classification of our survey. Lastly, a short survey by Maurer [17] included some selected studies proposing anonymity and privacy improvement methods, and compared nine approaches.

Our survey differs from the above-mentioned studies by (*i*) focusing on anonymity and privacy issues, (*ii*) including more studies and proposals and (*iii*) classifying these studies according to methods and outcomes. We believe that this survey will give insights to the researchers, who wish to study specifically anonymity and privacy issues in Bitcoin-like digital cash systems, at their starting point. We examined quite a number of studies related to Bitcoin and similar cash systems, especially the ones that utilize a blockchain structure. Among these studies; while some of them investigate security deficits, attack types, double-spending topics

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

or perform the evaluation of Bitcoin on reliability, scalability, stability or similar practicability issues, we included only the ones that are related to anonymity and privacy. For instance, Ethereum [18], [19], which is the second widely used digital currency [8] utilizing smart contracts, is not covered in our analysis and taxonomies since it does not focus on improving anonymity and privacy. Transactions in Ethereum are public [20] as in Bitcoin. Nevertheless, proposals for improving anonymity and privacy in Bitcoin can also be evaluated and used for improving anonymity and privacy in Ethereum [21] or similar cash systems. For instance, it was announced that integrating Zero Knowledge Succinct Non-interactive ARguments of Knowledge (ZK-SNARKs) [22], which are a special kind of Succinct Non-interactive ARgument of Knowledge (SNARK) [23], to Ethereum was an ongoing project as of January 2017 [24] and the integration was completed as of September 2017 [25].

In this study, we survey the studies related to anonymity and privacy in Bitcoin and similar digital cash systems in two main categories: (*i*) the studies that analyze the anonymity and privacy in Bitcoin, and (*ii*) the studies that propose improvements for anonymity and/or user privacy as an extension or alternative to Bitcoin. In this survey, the studies in these two major categories are examined with respective to several properties, divided into subcategories and presented visually. For the studies analyzing anonymity and privacy in Bitcoin, we also define characteristic properties and specify relationships of these properties with methods and outcomes. For the studies with anonymity and privacy improvements, we also show relationships among them. Moreover, we compare them from the performance perspective. We also summarize lessons learned and define a set of guidelines for designing an anonymity and privacy improvement for Bitcoin-like digital cash systems. Furthermore, we mention potential research topics for future research.

The organization of this survey is as follows. Section II gives background information about anonymity, privacy and Bitcoin. Section III details taxonomy of studies on anonymity and privacy analysis in Bitcoin. Section IV details taxonomy of papers on extensions and alternatives to Bitcoin with improvements on anonymity and privacy. Section V includes lessons learned, and Section VI gives the future research directions. Finally, we present the conclusion in Section VII.

II. Background

The background information for this study is given in the following three subsections; the first two subsections give brief information about anonymity, privacy, and Bitcoin. Then anonymity and privacy in Bitcoin and blockchain are explained in the last subsection.

A. Anonymity and Privacy

Anonymity and privacy are two concepts for which telling the difference may be difficult as Bradbury [26] mentioned, where privacy is hiding the context, and anonymity means hiding the owner of it. In daily life, generally, user privacy is sought more than anonymity, since personal data needs to be protected for proper usage. For instance, ownership information of a personal e-mail account can be known by everyone, but the content is restricted, protected and can be accessed by only the account owner using a password. Privacy is also essential in most systems and applications [27]–[29]. On the other side, anonymity is maybe the most important property that the criminals seek. The actions of criminals become usually public, but the actor aims his identity to remain unknown. With anonymity, holding someone accountable for an action becomes impossible [30]. However, there are some cases where anonymity is desired in daily life too. An example may be the applications, which are practiced in companies occasionally for the workplace evaluation. In these applications, personal opinions on a topic are gathered without identity information at an out-of-sight place. Then, the opinions are consolidated and announced publicly. Another well-known example is voting in free elections (secret ballot).

For anonymity, the objective is being unidentifiable and untraceable [31]. Ensuring true anonymity is difficult. Many applications claiming to be anonymous occur to have flaws, which leaks identity information. Mixing services [32], which are also called mixing networks (mixnets) or laundry services, are used for preventing tracing activities of messages through a network by including a sequence of intermediaries or a pool structure. However, they cause computation and communication overheads [33] or may be unreliable. Also, anonymization services, which use onion routing [34], are widely employed for hiding identity by addressing the issue of IP tracking. Even TOR [35], which is one of the most successful anonymity networks, is known to have flaws [26], [36]. Besides, these kinds of mixing services may be blocked by some websites or applications, so they are not always utilizable. One of the most dominant factors that prevent true anonymity is meta-data. In systems that consist of electronic transactions, meta-data of the transactions, e.g., log data, may lead to identities when handled with an analytical and holistic approach. For instance, IP addresses or timing of transactions are the data which can be utilized. One well-known example for this case is AOL releasing an "anonymized" search history for researchers, which then caused unexpected and undesirable results as researchers could find out the identity of individuals. One disclosed identity was Thelma Arnold divulged with her research history which shows her personal interests [26].

Anonymity and privacy usually come with a price. On the one hand, in general, systems which aim to provide anonymity and privacy require more resources in space, time or computational power, since extra work is done. One the other hand, the users need to pay more to become anonymous and private. For instance, the mobile applications that bring drivers and passengers together, i.e., Uber and similar applications following it, provide cheap riding service compared to taxicabs. However, users need to reveal their identities to use these applications, since the driver and the passenger rate each other about the ride and each ride is logged. This requirement causes the loss of anonymity and privacy which is provided in a regular and more expensive taxicab service. This loss may have negative consequences and a recent incident occurred in Turkey can be given as an example. According to [37], a passenger using

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

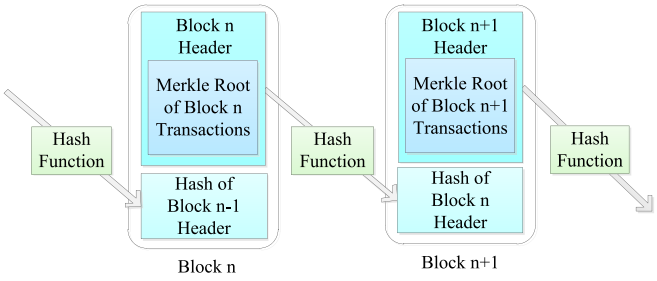

Fig. 1. Simplified version of blockchain.

a local mobile application in Turkey was allegedly assaulted by a spiteful driver who did not like the rating and comment of the passenger. In this incident, the driver hung around where he picked and left the passenger during a couple of days and finally caught another ride request of the same passenger. In this way, the spiteful driver was able to locate this passenger. If the user privacy and anonymity were preserved in this local Turkish application, the driver would not be able to learn any personal information of the passenger and would not be able to pinpoint him again.

B. Bitcoin

Bitcoin is a distributed, peer-to-peer (P2P) digital currency where no central authority exists. *Bitcoin* is the unit of the currency, and it is shortened as *BTC*. *Satoshi* is the smallest unit of the currency, and it is equal to a one hundred millionth of a single bitcoin (0.00000001 BTC). Bitcoins can be transferred from one address to another address. A transaction is a transfer of bitcoins and/or satoshis. Transaction management and issuance of bitcoins are performed jointly by the peers in the network.

Bitcoin uses public key cryptography. The address of a user is the hash of the public key of the user. The user can spend his bitcoins by using his private key to sign in a transaction. As a matter of fact, a bitcoin is a chain of digital signatures. A user can pass a bitcoin to another user by digitally signing hash of the previous transaction by his private key and includes the public key of the new owner in the transaction. New owner verifies signatures to verify the chain of ownership.

1) Blockchain: The blockchain is the general ledger of Bitcoin; it is the public record of all transactions, which are shared between all users and used to verify transactions. Blockchain consists of blocks. A block contains and confirms a part of new waiting transactions. Confirmation means a transaction getting processed by the network and being added to the blockchain. Transactions at each block are hashed, paired and hashed again until a single hash is obtained, which is the Merkle root [38]. Merkle root is stored in block header. Each block also includes hash of previous block header, which results in a chain of blocks. The basic structure of blockchain is given in Fig. 1.

2) Transactions: Each transaction has at least one input and one output that include address and amount information. In the input, a user can use bitcoins, which was received as an output in one or more transactions previously. As a result, flow of bitcoins between transactions also forms a chain structure. An output, which is not spent by an input, stays as Unspent Transaction Output (UTXO) until it is spent. The sum of all UTXOs assigned to a user determines the balance of the user. For example, if we say that one has 10 bitcoins, this means that he has 10 bitcoins waiting in one or more UTXOs assigned to him. An example illustration of this chain structure is shown in Fig. 2. The difference of sum of outputs and sum of inputs in a transaction corresponds to the transaction fee. Transaction fees of all transactions in a block are earned by the miner, i.e., the user who generated that block.

The conditions, which allow the transfer of bitcoins that are held in an output of a transaction to an input of another transaction, are specified by a script, written in a simple non-Turing-complete scripting language. In Fig. 3, an output segment of a transaction and the corresponding input segment are shown in more detail. The output of transaction n goes to the input of transaction $n+1$. The one who can satisfy the conditions of pubkey script in the output segment of former transaction gets ownership of bitcoins in the specified amount. The data parameters, which satisfy the conditionals in the pubkey script, are provided in the signature script in the input of the latter transaction to spend these bitcoins. For example, if Alice sends bitcoins to Bob in transaction n, in the output of transaction n, Alice indicates Bob; i.e., in order to assign these bitcoins, she mentions Bob's public key in pubkey script segment. When Bob wants to spend these bitcoins in transaction $n+1$, Bob has to represent himself in the input, therefore he uses his signature (private key) in the signature script. Elliptic Curve Digital Signature Algorithm (ECDSA) [40] is used with the secp256k1 curve for digital signatures.

3) Change Addresses: When a user wishes to spend an output of a transaction which is owned by him, he has to use all of it. This is an important difference between Bitcoin transactions and regular bank transactions. For example, in a regular bank account, if a user has 50 dollars in his account, he can use just 20 of it in a payment. However, in Bitcoin, if a user has 10 BTC as a UTXO assigned to him, he has to use all of this 10 BTC in a transaction. If the payment amount, which is indicated in one output of the transaction, is less than the amount in a UTXO, the user should state a second output address belonging to him to get back the change, if he does not want to give the remaining amount as the transaction fee. As the change address, the user can use an old address or generate a new address to use. It is suggested to use a new address at each transaction to reduce traceability and improve anonymity.

4) Mining and Incentives: Bitcoin transactions are broadcast to the network. Since the transactions are public, nodes running Bitcoin software checks their validity. Then in the *mining* process, they form a new block that contains new transactions. This new block is added to the latest copy of the blockchain and broadcasted to other nodes. The process of adding a block is called *mining* since each block comes with a reward. In each block generation, new bitcoins are issued and assigned to the creator of the block. Block generation reward halves at every 210 thousand blocks. It was 50 BTC at the beginning, then decreased to 25 BTC and it is 12.5 BTC

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

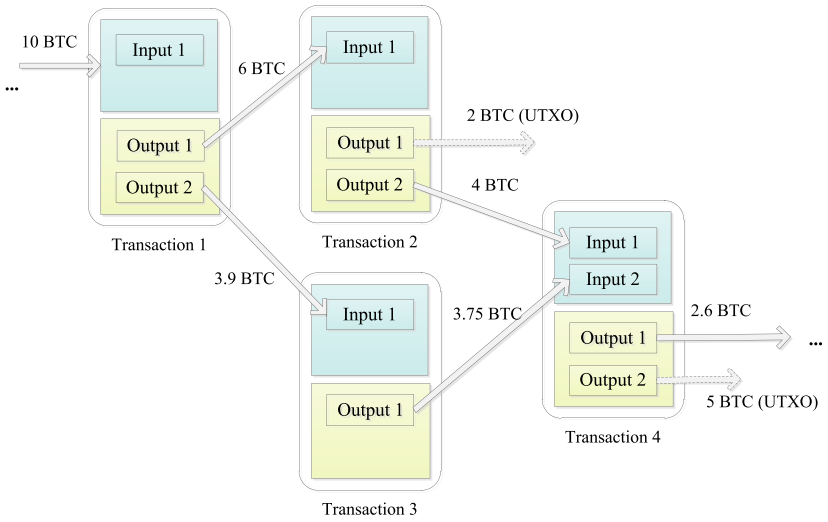

Fig. 2. A sample flow of bitcoins from transactions to transactions.

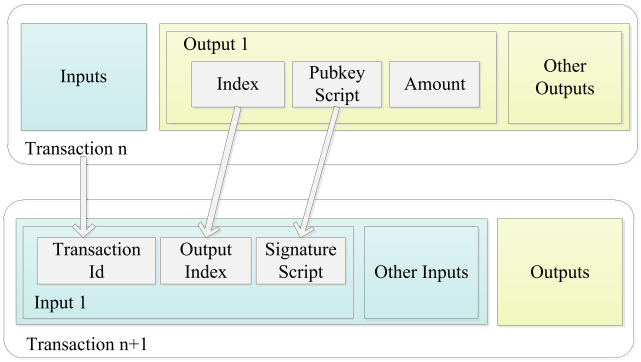

Fig. 3. Output and input segments of two related transactions [39].

since July 2016. Block generation reward is implemented by putting a new, special transaction as the first transaction in the block where the payee is the creator of the block. This first transaction in the block is called *coinbase transaction*. There is a race in this mining process to get the reward. The mining process also acts as an incentive for nodes to support the network and puts new bitcoins into circulation.

Creator of a block is also rewarded with the transaction fees of the block. A transaction has a transaction fee if the output value of a transaction is greater than its input value. Payers include fees in order to get their transactions processed quicker; miners may select to process transactions with fees. Maximum numbers of bitcoins are determined to be 21 million. Therefore, after that number is reached, the only incentive to mine will be the transaction fees. These incentives increase the possibility of the nodes behaving honestly.

The longest blockchain is recognized as the actual and the latest blockchain. All miner nodes work for generating another block to add to this latest copy. Transactions receive a confirmation when they are included in a block, and a transaction is confirmed again each time another block is added to the blockchain after the block of the transaction.

It should be noted that UTXO of a coinbase transaction cannot be used as an input, which means cannot be spent, for at least 100 blocks. This is for guaranteeing that a block reward is not spent until the permanence of the block in the blockchain becomes absolute. This rule is required due to the possibility of having of blockchain forks.

5) Proof-of-Work: In a block header, a nonce value is included, which is used as the proof-of-work for creating the block. Proof-of-work is a system to prevent Denial-of-Service (DoS) attacks and other system misuses. This system requires a user to show that he performed some work and spent some effort, i.e., processing time, to complete a task. However, the proof can be easily verified. A proof-of-work is like a puzzle, takes some time and effort to solve, but it can be verified easily. Bitcoin uses a proof-of-work algorithm named *Hashcash* [41]. In Hashcash, a hash value, which begins with a specific number of zeroes, is required. Since this is a specific requirement and cannot be obtained directly, it shows that the user spends some effort and time. Hashcash can be implemented by incrementing a nonce value until it provides the requirement of a specific number of zeroes at the beginning of its hash. An example of the Hashcash implementation is given in Fig. 4. In this example, the required number of zeroes is determined as four in hexadecimal. Nonce value is added to the end of the block data. This nonce value, starting from zero, is incremented until the hash begins with four zeroes. It takes 1,307 tries to obtain a valid hash value.

In this scheme, average work required is exponential in the number of required zero bits, but it can be verified by computing a single hash value. For instance, if the required number of zeroes is 20 and SHA-256 is used as the hash function, out of 2^{256} possible hash values, there are 2^{236} hash values that satisfy this criterion. The possibility of randomly selecting a number that will have 20 zeroes as the beginning of the hash is 1 in 2^{20}. Thus, one has to try $2^{20}/2 = 2^{19}$ values on the average to find such a hash value.

While miners compete for adding a block to the blockchain, they actually race for finding a proof-of-work for a block of transactions. By always accepting the longest chain as the actual and the most recent blockchain, the greatest proof-of-work computation is guaranteed. This means if the majority

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

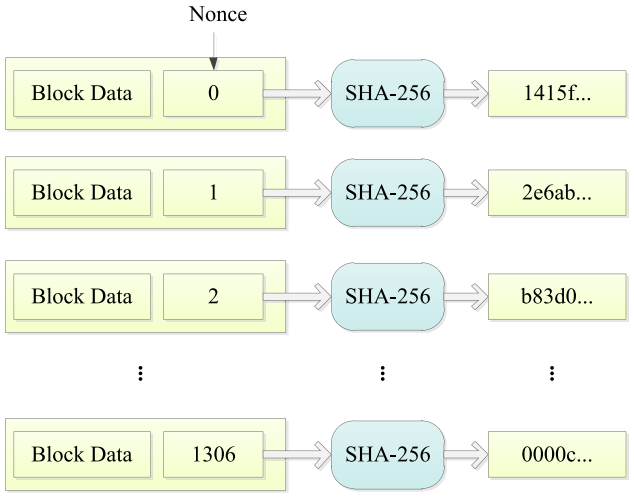

Fig. 4. Hashcash example.

of nodes in the network are honest, then the chain formed by these honest nodes is the longest and beat any other alternative, competing chains. For example, to modify a past block, finding proof-of-work of the block and the blocks after that block must be achieved again. Also, the work and the chain of the honest nodes must be outperformed, i.e., a longer chain must be obtained. The probability of an attacker to be successful decreases exponentially as the successive blocks are added.

The number of zero bits determining the proof-of-work difficulty is named as *difficulty target*. Improvements in the hardware speed and changes in the number of running nodes in the network affect the generation rate of blocks. Therefore, there is a need for difficulty retargeting and the difficulty target is adjusted according to the block generation rate (average number of blocks per hour), i.e., if blocks get generated too fast, the difficulty increases by increasing the required number of zeroes. The average number of blocks per hour is determined to be 6, so creating a block should take 10 minutes on average. In order to achieve this, every 2,016 blocks, the network calculates the number of seconds to generate these 2,016 blocks by using timestamps at block headers. This value is expected to be 1,209,600 seconds, corresponding to two weeks. If it took less than two weeks to generate these 2,016 blocks, then the difficulty target is increased proportionally, and if it took more than two weeks, then the difficulty target is decreased proportionally.

6) Double-Spending: The most known problem for digital currencies is the *double-spending* problem, in which a malicious user tries to spend to two different payees with the same money. To prevent double-spending, it must be ensured that a digital coin is not spent twice by its owner. Before Bitcoin, this problem is solved by using a central authority which approves transactions. However, in Bitcoin, if someone tries to spend the same bitcoin twice, blockchain acts as the single source of verification, and the network does not accept to add the second transaction to the blockchain. This is achieved by nodes following *consensus rules*. These are the rules that nodes in the network follow to maintain consensus, i.e., to reach an agreement on having same blocks in their blockchain. These rules are also called *validation rules* since transactions and blocks are validated according to these rules, and a block violating the consensus rules is rejected. Examples of Bitcoin consensus rules are as follows; (i) signatures must be correct for the bitcoins being spent in transactions and (ii) the maximum number of bitcoins that can be created in a block is limited (12.5 BTC as of March 2017).

The proof-of-work (PoW) scheme in the mining process is used by Bitcoin to reach consensus on the blockchain and presents a solution [42] to the Byzantine Generals Problem [43]. Miners can include different transactions in their blocks while racing for adding a new block to the blockchain; therefore, it is possible that two miners come up with two different new blocks at the same time and broadcast to the rest of the network. This results in a forked blockchain and obligates the network to reach a consensus on which block to add to the blockchain. In this case, miners save both blockchains but select one of them and try to find a new block to add that chain. Meanwhile, if a miner receives a new block from the network for one of the chains, he discards the shorter chain and continues working for adding a new block to the longer chain. So the blockchain provides the structure as a single history of the order in which transactions were processed and assures the integrity of the system without a central authority. This is achieved by recording information of all transactions which are impossible to forge but at the same time quickly verifiable. In order to add a block to the blockchain, a hash (SHA-256) of a block of items (transactions) with their timestamp value is taken which proves the existence of data at that time, and the order between transactions is established. While taking the hash, the hash of all previous blocks is also included as a chain, so each block (and its timestamp) supports the integrity of the previous blocks. Thus, modification of a block requires modification of all the blocks coming afterward. This modification requirement requires a huge amount of processing power due to the proof-of-work structure. To change a transaction which happened 60 minutes (6 blocks) ago, e.g., to remove information that spending some bitcoins for double-spending, one has to change the record for that transaction and solve a new proof-of-work problem for that block (find a new nonce). Then he has to construct an alternative chain which goes forward, by solving a new proof-of-work problem for each block and surpass the actual chain by forming a longer chain. This can be achieved if and only if the malicious user has more computing power than the sum of all other miners' powers and this is known as a *majority attack*; otherwise, he cannot surpass the actual chain. It is suggested to wait for 6 confirmations for a high-value bitcoin transfer, which means that 5 additional blocks should be added after the block that contains the transaction [44].

7) P2P Network: Peers connect to each other over an unencrypted TCP channel. When a peer first wishes to enter the network, some DNS servers, which are called DNS seeds, are queried. These DNS seeds are hardcoded in Bitcoin clients to find active peers. The response includes IP addresses of the peers that accept new incoming connections. To connect a peer, a *version* message is sent, including version number, block, and the current time of the sender peer. Receiver

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

peer replies this message with its *version* message. Then both nodes send *verack* message to acknowledge that the connection has been established. After a peer enters the network, peer discovery is based on an address propagation mechanism, where peers can request IP addresses from each other with *getaddr* messages, and send their IP address lists to other peers with *addr* messages. Each address has a timestamp which shows its freshness. Peers can have a total of 125 connections, where 8 of them are outgoing connections, and 117 of them are incoming connections, except peers behind NAT or firewall; they can have only outgoing connections and cannot accept incoming connections. Each peer stores a list of IP addresses of their connections. When a new block is generated by a miner, it is broadcasted by the miner to its peers. Receiving peers validate the block and then forward to their peers. When a peer forms a transaction, then it sends this message to its connected peers. A transaction is sent to a peer by first sending an *inv* message. If the receiver peer replies with a *getdata* response, then the transaction is sent using *tx* message. The transaction continues to be propagated between peers in the same manner if it is a valid transaction. A reputation based protocol exist, where each peer keeps a penalty score for every connection and increases these penalty scores for the connections that send faulty messages. The peer bans a connection for 24 hours when its penalty score reaches a threshold.

8) Summary of the Process: As a summary, the process can be defined as follows:

- New transactions are broadcast to all nodes.
- Each miner node collects new transactions into a block.
- Each miner node works on finding a proof-of-work for its block.
- When a miner finds a proof-of-work, it broadcasts the blockchain with the added block to all nodes.
- Other miner nodes accept the addition only if the block and all transactions in the block are valid.
- Other miner nodes show their acceptance by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.
- Miner nodes consider the longest chain and keep working on extending it.
- If two miner nodes broadcast different versions of the blockchain with a valid new block addition at the same time, miners work on the first one they received, but also keep the other chain for the possibility of it becoming longer.
- When a proof-of-work is found for the next block, and one chain becomes longer, nodes that were working on the other chain discard that chain and switch to the longer one.

A brief technical background on Bitcoin is given herein. Unquestionably, the whole technical structure of Bitcoin is much deeper, more detailed information can be found given in Bitcoin Developer Guide [39].

C. Anonymity and Privacy in Bitcoin and Blockchain

In the traditional banking model, information about the parties in transactions are limitedly shared and secured by the trusted third parties, however the banks, which are trusted third parties, know everything about their customers. In Bitcoin, everything is transparent; all transactions are publicly announced. The only thing done for anonymity is to keep public keys anonymous, using pseudonyms for the addresses. Everyone can monitor that users transfer bitcoins to each other, but the real names are not provided, only the pseudonyms are used. It is similar to the stock exchange operations in that sense. Since pseudonyms are used in Bitcoin transactions, the general impression can be as Bitcoin provides anonymity. However, as mentioned explicitly by Bitcoin [44], it is not anonymous, and it is remarked as "probably the most transparent payment network in the World" [45]. All transactions are kept public where payers and payees are specified with their pseudonyms, which means all transactions of a Bitcoin address (pseudonym) can be seen. Blockchain does not keep balances, just the transactions; however, balance of an address can be calculated after obtaining all transactions related to this address. Nevertheless, since no other information about the users is stored, real identity of a user is not known, unless there is a need for revealing it, e.g., in order to receive services or goods or law enforcement. Even so, the real identity is not revealed to everyone unless the service provider announces publicly. So it can be said that Bitcoin users cannot stay completely anonymous, but can be considered as pseudonymous.

Except revealing identity to receive service or goods to merchant and/or payment processor, identity and address can be linked while trading bitcoins on exchange since exchanges may be subject to money laundering regulations. In this case, customers need to prove their identity to the exchange.

It can be said that provided level of privacy is determined by the behavior of a user while using Bitcoin. Some security countermeasures are suggested by Bitcoin [45], e.g., generating and using a new key pair (a new address) for each transaction. Because when a new key pair is generated, it cannot be linked to previous transactions of the user and therefore the number of bitcoins the user has cannot be learned. Another recommendation is to use separate wallets for different purposes. *Wallet* refers to the programs or files that are used for managing transactions and creating and managing Bitcoin addresses, public-private key pairs [39]. Transactions at different wallets cannot be linked and can stay isolated. Also, the first thing that has to be done for privacy is to be careful about not to disclose addresses. However, it should be noted that there might be cases where the address is disclosed, e.g., for receiving public donations, or for proving a payment is made in order to receive a good. Similarly, information about transactions, like amount, should not be disclosed since they may help to find addresses related to them.

Hosted wallet services know the addresses of the users who use them, because data of wallets are stored in servers owned by the wallet services. Additional information to use these wallet services, such as e-mail address or phone number, can help these services to link this information to identity. A list and ranking of the wallets according to the privacy level that

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

they provide as of March 2016 can be found in OBPP Bitcoin Wallet Privacy Rating Report [46].

Bitcoin also warns its users to take some issues into consideration of which users may not be aware. For instance, it should be known that IP address of a Bitcoin address can be logged by connecting to active nodes in the network and listening for transaction relays. Similarly, the Internet service provider (ISP) can intercept transaction messages that a user sends and figure out the addresses owned by him. IP addresses do not directly reveal identity, but they can be used to find it. These deanonymization methods are examined in more detail in Section III. IP addresses can be hidden by using anonymization services like TOR. There are also mixing services, which mix transactions of users by receiving and sending back the same amount using independent addresses and make impossible to trace activities of users. However, these services require trust to the service since the users actually transfer bitcoins to the service. Moreover, these services can log requests of users. Another reason which limits the usage of mixing services is that these services are inefficient for hiding large transactions, although they are effective while hiding small amounts. These services are examined in more detail in Section IV.

Bitcoin's blockchain was designed to be public, and it does not provide privacy per se. However, blockchain can be used for many purposes and in many sectors. Thus, studies focusing on general blockchain anonymity and privacy may also be surveyed, but this is not the focus of this survey. Nevertheless, we will just briefly mention some important blockchain privacy studies here. Enigma [47] is a peer-to-peer network in which different parties can jointly store data and run computations, at the same time keeping the data private. The computational model of Enigma is based on Secure Multiparty Computations. Data is split between nodes and data queries are computed distributedly. Blockchain serves as an unalterable log of events and manages identities and access control. Xu *et al.* [48] proposed a privacy respecting approach for blockchain-based sharing economy applications, which leverages a zero-knowledge scheme. There are also recent studies on privacy in blockchains of smart contracts. Ethereum [18], [19] is the first smart contract blockchain platform. However, all transactions in Ethereum were public, as in Bitcoin. As mentioned in Section I, ZK-SNARKs [23] can be utilized in Ethereum transaction since September 2017. Moreover, Ethereum platform enables setting up private and permissioned blockchains to improve privacy. For instance, Quorum [49], [50] is a private and permissioned Ethereum-based smart contract blockchain. Quorum supports both public and private transactions. Details of private transactions are revealed only to those party of the transactions. Symmetric encryption and hash functions are used to keep data private. A distinct private state database is stored at each node additional to the common public state database. Private contract code in a private transaction can only be executed by the nodes that are party to that transaction. Privacy is obtained in Hyperledger Fabric [51], another smart contract platform, similar to Quorum; by using hash functions and symmetric encryption.

III. Taxonomy of Studies on Anonymity and Privacy Analysis in Bitcoin

We classify methods of analyzing anonymity and privacy in Bitcoin that are described and used in the literature as given in Fig. 5. Essentially, analyzing anonymity and privacy is done through spending effort to achieve deanonymization and extract information that would impair privacy of users. Therefore, methods in the literature serve these purposes. For a method, while some studies may use the method, some studies may only mention the method but do not use it; therefore, for each method, studies that mentioned or applied the method are given respectively. Resulting outcomes are given in the bottom part of the figure. The outcomes and the methods are explained in detail in the first two subsections. The third subsection analyzes the studies according to other characteristic properties determined to examine these studies.

A. Outcomes

Outcomes are actually potential aims to be achieved after analyses. There are five outcomes of analyzing anonymity and privacy in Bitcoin. Each outcome is described in detail in the following.

1) Discovering Bitcoin Addresses: Possible Bitcoin address of a person or an entity is discovered starting from an identity information, such as name and surname of a person, or name of a company.

2) Discovering Identities: Possible identity information, such as name and surname of a person or company name is obtained starting from a Bitcoin address.

3) Mapping Bitcoin Addresses to IP Addresses: Bitcoin addresses are mapped to possible IP addresses where the transactions are generated.

4) Linking Bitcoin Addresses: Bitcoin users are suggested to use new Bitcoin address each time they receive a new payment [45]. Therefore a user can have more than one Bitcoin address. Addresses expected to belong to the same user are linked together in this outcome.

5) Mapping Bitcoin Addresses to Geo-Locations: Information about the physical location of a user is obtained starting from Bitcoin address.

Transition can happen between these outcomes. For instance, a Bitcoin address belonging to a person can be discovered, and then this address can be linked to his other Bitcoin addresses. Similarly, a Bitcoin address can be mapped to the IP address, and then this IP address can be used to discover identity or geo-location of the user that owns the Bitcoin address.

B. Methods

Each method is described in detail, and the related studies for each method are given in the following subsections.

1) Transacting: By transacting with other users, e.g., purchasing of goods and services, Bitcoin address of the other end is learned. A buyer must know Bitcoin address of a seller in order to make a payment to the seller, so a seller must share his Bitcoin address if he wants to receive payments. Therefore, one can act as a buyer and learn Bitcoin addresses

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

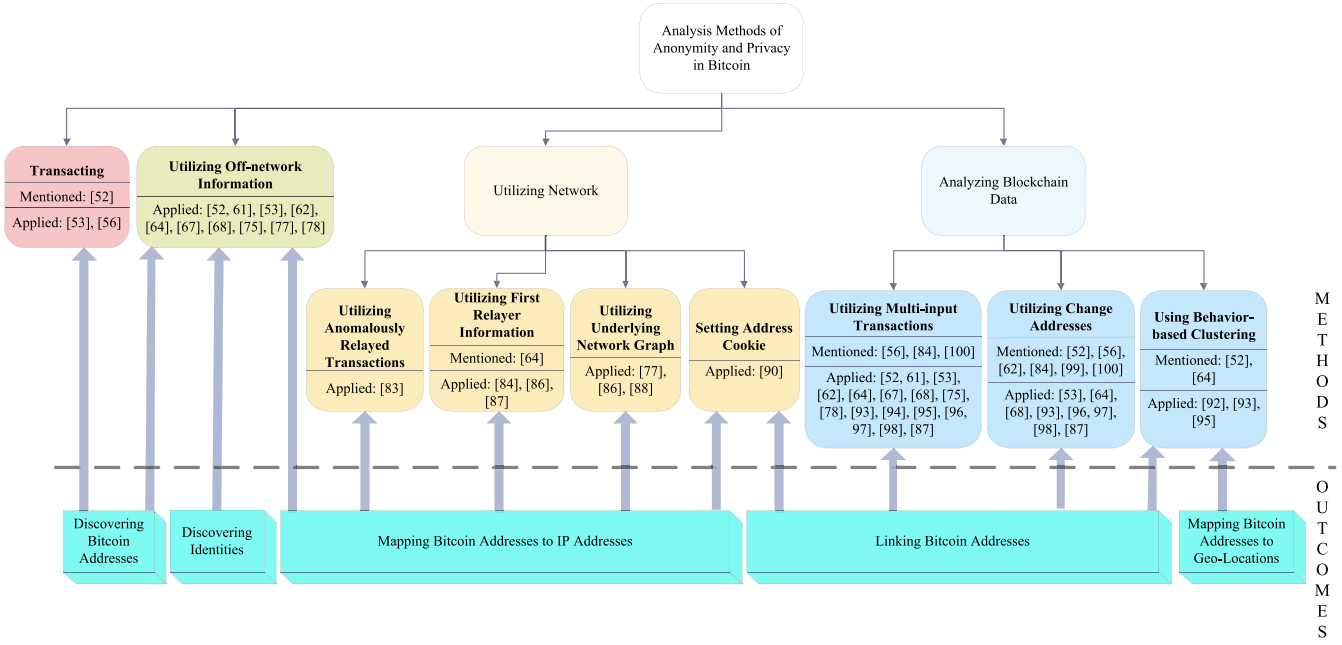

Fig. 5. Taxonomy of methods of analyzing anonymity and privacy in Bitcoin and the outcomes.

of parties that he would like to know, assuming that these parties are involved in sales activities. Transacting method means actively participating in the network and may also include using marked coins or operating a money laundry service as Reid and Harrigan [52] stated. Meiklejohn *et al.* [53] used transacting method and named it as *re-identification attack*. Their approach was opening accounts and making purchases from well-known Bitcoin merchants and service providers like Mt. Gox [54] and Silk Road [55]. They engaged in 344 transactions with 87 different services including mining pools, wallet services, bank exchanges, non-bank exchanges, vendors, gambling sites, and miscellaneous services. As a result, they could identify 1,070 Bitcoin addresses. Transacting can also be used to understand the mode of operation of mixing services, i.e., anonymization services. The effectiveness of anonymization by these services can be examined by tracing performed transactions. Möser *et al.* [56] selected three mixing services, namely Bitcoin Fog [57], BitLaundry [58] and the Send Shared functionality of blockchain.info [59] and analyzed them using this approach. These services are called as *money laundering* tools in the study. They traced anonymized transactions to their probe accounts in the blockchain. They found that Bitcoin Fog and blockchain.info were successful at anonymizing; however, they were able to link the input and output transactions of BitLaundry.

2) Utilizing Off-Network Information: Publicly available off-network data sources, which are obtained externally (out of Bitcoin network and blockchain), can be used to discover identities belonging to Bitcoin addresses, or vice versa. Donation websites that include information related to Bitcoin addresses to prevent service abuses can be given as an example of these data sources. Also, users can voluntarily disclose Bitcoin addresses in forums. In addition, large and highly active entities are publicly recognized on the website blockchain.info [60]. Off-network information from this website can also be used to obtain IP address belonging to a Bitcoin address that initiates the transaction.

Reid and Harrigan [52], [61] utilized public donation websites which publish IP and public key information together with the giveaways, like The Bitcoin Faucet and voluntary disclosure of public keys in forums. They identified some entities related to an alleged theft of 25,000 BTC, by using off-network information. They also stated that order books of Bitcoin exchanges can be used to map Bitcoin addresses and transactions to the exchanges [52]. The ability to link addresses of a user does not reveal the identity of the user. However, if one of the addresses in a linking is combined with external information that is leading to the identity of the user, all activities of the user become disclosed. Ron and Shamir [62] gave the following example: Since WikiLeaks [63] publicly advertised one of its addresses for donations, they could estimate that at least 83 addresses were owned by WikiLeaks which was involved in at least 1,088 transactions and total balance in all these addresses was 2,605.25 BTCs.

Ortega [64] provided scripts for linking Bitcoin addresses to the identities from information provided in forums by users. He collected 4,000 Bitcoin addresses in a two weeks period from Bitcoin forum [65], which is a popular online forum for Bitcoin users. In this forum, users can declare a real-world location and a Bitcoin address. Ortega assigned these 4,000 Bitcoin addresses to 1,825 different users since some users include several different addresses in their posts. Meiklejohn *et al.* [53] gathered over 5,000 addresses from address tags of blockchain.info [66] which collects addresses that are provided in users' signatures in Bitcoin forums and also self-submitted tags. In addition, they searched through the Bitcoin forums, in particular, bitcointalk. They stated that this method is less reliable than directly transacting. Therefore, they used the addresses that they could gain some confidence through manual examination in their further analyzes.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

Fleder *et al.* [67] scraped Bitcoin addresses from bitcointalk forum signatures and managed to detect 2,322 users with 2,404 addresses in just under 30 hours. They also studied on identifying a transaction by examining every transaction in the blockchain after taking rough information regarding transaction time and value, and then creating time and value windows by varying amounts. BitIodine [68], which is an open blockchain analysis framework that is introduced by Spagnuolo *et al.*, uses a set of Web scrapers that automatically collect and update lists of Bitcoin addresses belonging to known identities. Spagnuolo *et al.* used signatures and databases information from forums, namely bitcointalk forum and Bitcoin-OTC marketplace [69]. They also used information of physical coins created by Casascius [70]. Another information they used was known scammers, by automatically identifying users that have significant negative feedback on the Bitcoin-OTC and bitcointalk trust system. Shareholders in BitFunder, which was a stock exchange that is now closed [71], was also another source of information. The authors stated that data from address tags of blockchain.info can be used with a semi-automatic approach, as well. They also got historical data about trades of Bitcoin by using Mt. Gox trading APIs to detect interesting flows of the Bitcoin economy. BitIodine was tested on several real-world use cases in the study, like identification of an address that is likely to be owned by Silk Road or investigating the CryptoLocker [72] ransomware. It should be noted that BitIodine is no longer actively developed and does not work with the recent Bitcoin blockchain [73]. Instead, rusty-blockparser [74] which is a multi-threaded parser for Bitcoin-based blockchains is supported.

Baumann *et al.* [75] got IP address information belonging to Bitcoin addresses from the site blockchain.info. They could link Bitcoin addresses that are owned by Mt. Gox with this method. However, they noted that many IP addresses cannot be associated directly with the real user since they just reveal information about the last gateway before entering the blockchain. Bitnodes [76] was mentioned as the source for getting information about the users that are not using hosting services. Biryukov *et al.* [77] used data from Bitnodes, which produces a list of running Bitcoin servers every five minutes in order to estimate the probability of an entry node going off-line. Lischke and Fabian [78] gathered IP address data from the websites blockchain.info and ipinfo.io [79]. ipinfo.io provides information about the geo location, hostname, or organization related to the IP. They gathered over 223,000 distinct IP addresses that were used in around 15.8 million transactions. Other sources that they used for IP addresses are torstatus.blutmagie.de [80], dan.me.uk/torlist [81], vpngeeks.com/proxylist [82]. They also got Mt. Gox daily trade data from Bitcoin Charts website [9].

3) Utilizing Network: By analyzing Bitcoin network traffic or using network infrastructure, information about transactions can be obtained. *Utilizing anomalously relayed transactions*, *utilizing entry nodes*, *utilizing first relayer information* and *setting address cookie* for user fingerprinting are the analysis methods which utilize the Bitcoin network. Date interval for each study that makes analysis by utilizing the network is given in Fig. 6.

a) Utilizing anomalously relayed transactions: Sending of a message by a peer to other peers is also called relay. By analyzing Bitcoin network traffic and transaction message relays, abnormal relay patterns can be defined, such as a transaction being relayed by a single person or a transaction being rerelayed (relayed more than once) by at least one user. Transactions matching these patterns can be used to map Bitcoin addresses to IP addresses. Koshy *et al.* [83] were inspired by the idea of using P2P network information which was introduced by Kaminsky [84]. They proposed this method in their study which was the first to map Bitcoin addresses to IP addresses. They built a Bitcoin client called CoinSeer, which was designed to collect data. By using CoinSeer, they created an outbound connection to every listening peer whose IP address was advertised on the Bitcoin network. They maintained more connections than blockchain.info which is stated to be only other Bitcoin superclient at the time. They analyzed network data between 24th July 2012 and 2nd January 2013. After analyzing 5 months of network data, they classified distinct transaction relay patterns. Their relay patterns consisted of one normally relayed transaction pattern (multi-relayer and non-rerelayed) and three anomalously relayed transaction patterns (single-relayer, multi-relayer and single rerelayer, multi-relayer and multi-rerelayer). Then they designed heuristics to map Bitcoin addresses to IP addresses, and they were able to map between 252 and 1,162 Bitcoin addresses to IP addresses depending on the support and confidence parameters. However, one limitation of this approach is relying on anomalously relayed transactions, since they stated that normal transaction traffic is very difficult to deanonymize. Another limitation of the method is allowing to learn only IP addresses of the servers and transactions sent through anonymization services like TOR or by using wallet services would be assigned to incorrect owners.

b) Utilizing first relayer information: When connected to every node in the Bitcoin network, it can be assumed that the first node to inform of a transaction is the source, i.e., the owner of it. P2P network and relays were introduced as another source of data for deanonymization firstly by Kaminsky [84], where he proposed utilizing first relayer information. He developed a tool named Blitcoin [85] that uses this idea for deanonymizing one end of the transaction. This method was also mentioned and accepted by Ortega [64]. However, using this heuristic for normally relayed transactions was found to be ineffective compared to utilizing anomalously relayed transactions by Koshy *et al.* [83].

Fanti and Viswanath [86] focused on dissemination of transactions in Bitcoin P2P network and examined two network protocols used in Bitcoin; trickle spreading (pre-2015) and diffusion spreading (post-2015). They tried to detect IP addresses that initiated transactions by using an eavesdropper adversary, which just listens to all related messages on the network. They utilized first-relayer information as the deanonymization method in both protocols. They analyzed the probability of detecting source IP address and observed that this probability is lower in diffusion spreading than in trickle spreading. This

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

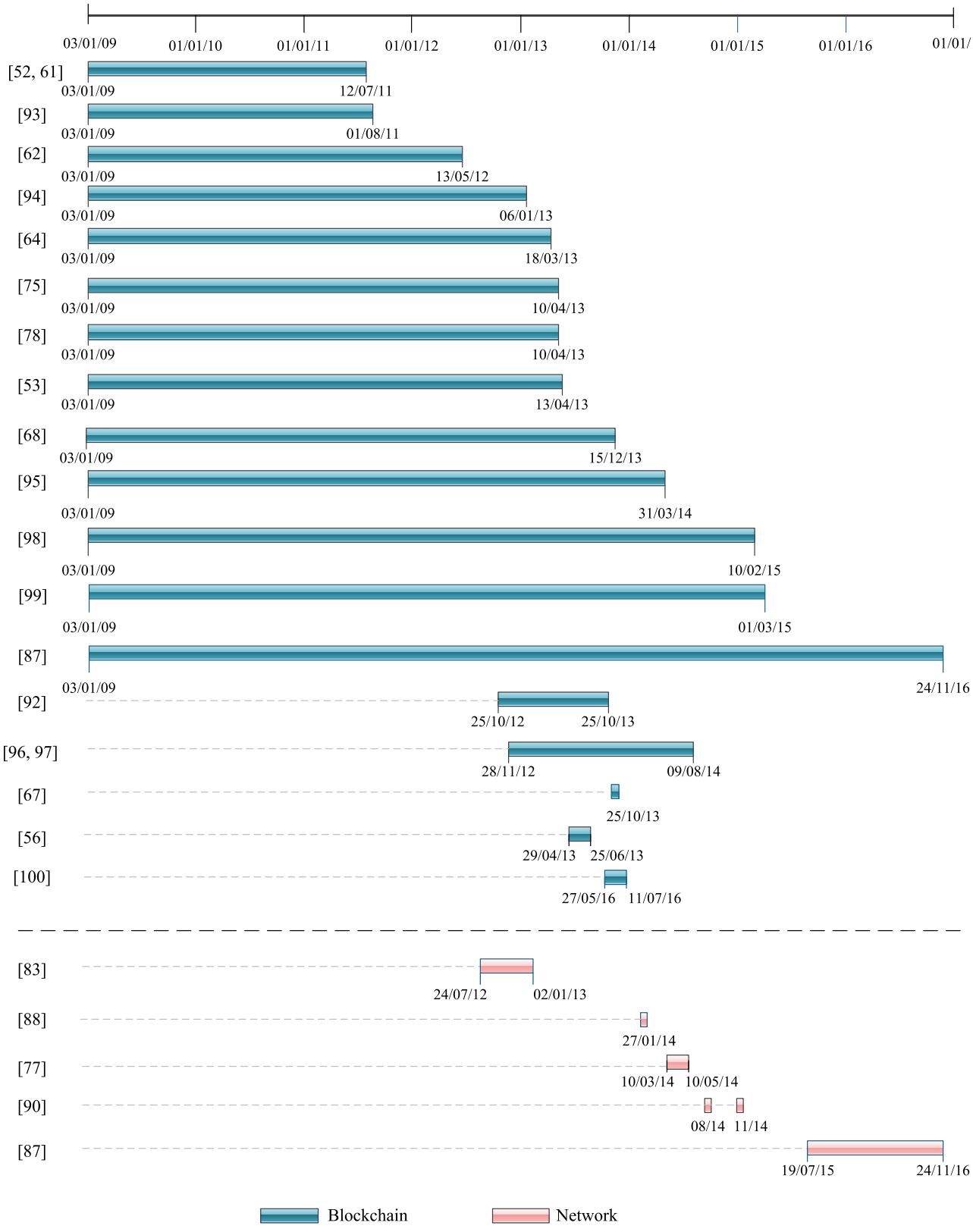

Fig. 6. Date intervals of the analyses.

shows that diffusion spreading is better than trickle spreading in anonymization. However, they stated that the difference is small and both protocols have similar and poor anonymity properties. Although they mentioned that they used a snapshot of the real Bitcoin network from 2015, the dates are not stated clearly. Therefore, this study is not included in Fig. 6. Neudecker and Hartenstein [87] utilized first relayer information and applied several heuristics to reduce the number of

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

false mappings. The main aim of their study was investigating the effect of network information on clustering Bitcoin addresses using blockchain data. Their results showed that network information does not facilitate address clustering for the majority of users.

c) Utilizing underlying network graph: Bitcoin clients can be identified by utilizing underlying P2P network graph.

Biryukov *et al.* [77] introduced utilizing entry nodes method. Entry nodes are the nodes that a Bitcoin client connects to. This information can be retrieved for a node when it connects to the network and can be used to identify the origins (namely the owners) of the transactions and map Bitcoin addresses to IP addresses. Biryukov *et al.* targeted the clients, i.e., peers behind NAT or firewalls of their ISPs, and could distinguish the nodes sharing same public IP. Their method first finds entry nodes of clients, then listens to servers and maps transactions to entry nodes and then to clients. Their method can link Bitcoin addresses which seem to be unrelated by analyzing blockchain data and also not limited to the anomalously relayed transactions. They collected statistics from their Bitcoin peer for 60 days, which are between 10th March 2014 and 10th May 2014, and collected information about 61,395 connections in total. They could identify 11% of transactions in the Bitcoin test network with a few GB of storage and no more than 50 connections to each Bitcoin server. They also showed how to discard TOR or other anonymization services by exploiting anti-DoS countermeasure of the Bitcoin network. This results in prohibiting Bitcoin servers to accept connections via TOR and other anonymity services and clients using their actual IP addresses when connecting to other peers. However, this attack is quite noticeable.

Fanti and Viswanath [86] utilized the underlying graph structure as the second method. They used information of ordering of the nodes according to receival of transactions in trickle spreading, and they used centrality information in diffusion spreading according to the observed timestamps. Feld *et al.* [88] connected peers, collected IP addresses from each peer and connected the returned addresses recursively. Then they determined autonomous system and country for each client using GeoLite databases [89]. They grouped 10,500 peers to 1,700 different autonomous systems.

d) Setting address cookie: Different transactions and Bitcoin addresses of the same user and IP addresses can be linked by setting an address cookie on the user's computer. The user can be fingerprinted simply by checking this address cookie. This method does not require blockchain analysis and is based on Bitcoin's peer discovery mechanism. Since Bitcoin peers get addresses from other peers, a unique combination of fake addresses, which would behave as a fingerprint, can be sent to a peer to fingerprint him. The peer stores these addresses. The next time he connects, his address database can be queried, and the user is identified if the fingerprint addresses (in the address cookie) are present. This method was proposed by Biryukov and Pustogarov [90] to correlate the same user across different sessions, even if he uses anonymization services like TOR or multiple proxies. It was stated that if the user later connects to the Bitcoin network directly, without using this kind of services, his IP address would be revealed and linked to his fingerprint since the cookie would be still present. This results in deanonymizing his all transactions sent previously through TOR. They queried address databases of reachable Bitcoin peers in order to discover ratio of the onion address type, which is the only address type that is accepted by other peers when using TOR. During August 2014, they discovered that among 1,153,586 unique addresses, only 228 were onion addresses, and only 39 of them were actually online. In November 2014, they repeated the experiment and among 737,314 unique addresses, 252 were onion addresses and 46 were online. Also, in November 2014, they queried running Bitcoin servers to estimate the probability that a cookie address, that was set, was preserved as the experiment for the stability of the cookie. They received 4,941,815 address-timestamp pairs, among which only 303,049 of the addresses were unique. This showed that only about 6% of the addresses received by a client are not already contained in his database. It was stated that it is almost guaranteed that a cookie will not be damaged within the first 3 hours according to the experiments. They also set a cookie consisting of 100 addresses and monitored the stability of the cookie. The experiment showed that even after 8 hours, one-third of the cookie remained in the user's address database and made possible to identify the user. It was stated that there were about 7,000 Bitcoin servers at the time. According to this, one needs about 90 peers to be one of the entry nodes within 8 hours and update the cookie. They estimated the cost for this as less than 650 USD per month. Finally, encrypting and authenticating Bitcoin traffic was proposed as the countermeasure to this user fingerprinting.

4) Analyzing Blockchain Data: Since entire transaction history is publicly available in the blockchain, flow of bitcoins between Bitcoin addresses is traceable. Blockchain data can be gathered using APIs of Bitcoin clients. Bitcoin Core [91], which is the official Bitcoin client, and blockchain.info are the two well-known and widely used clients. Reid and Harrigan [52], [61] were first to analyze blockchain for anonymity and privacy of Bitcoin. They introduced two network structures, transaction and user networks, which are also used and utilized in subsequent studies extensively. The flow of bitcoins between transactions over time is shown in the transaction network and flow of bitcoins between users over time is depicted in the user network. Constructing transaction network from blockchain is straightforward; transactions are represented as nodes and flow of bitcoins are represented as directed edges with amount and timestamp information. An output of a source node becomes input to a target node.

Fig. 7 shows an example sub-network of transactions. $t1$ has one input and two outputs. It was performed on 2nd February 2017, and one of its outputs transferred 0.5 BTC. $t2$is a transaction with two inputs and one output. It was performed on 30th April 2017, and the output transferred 0.3 BTC. $t3$ is a transaction with one input and one output. It was performed on 10th May 2017, and the output transferred 0.2 BTC. Finally, $t4$ is a transaction with three inputs and one output. It was performed on the 23th May 2017. All inputs come from the outputs of $t1$, $t2$, and $t3$, which are mentioned above. The output of $t4$ is 1 BTC; equal to the sum of its inputs.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

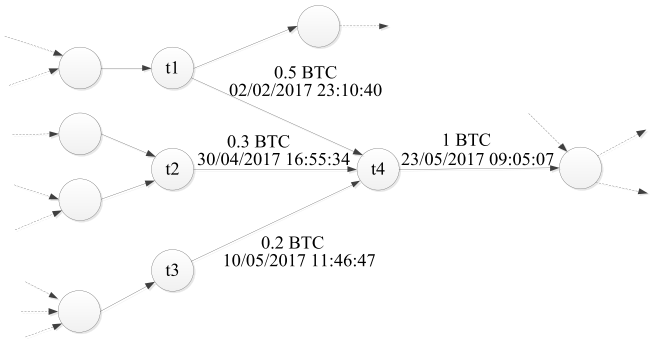

Fig. 7. Example sub-network of transactions.

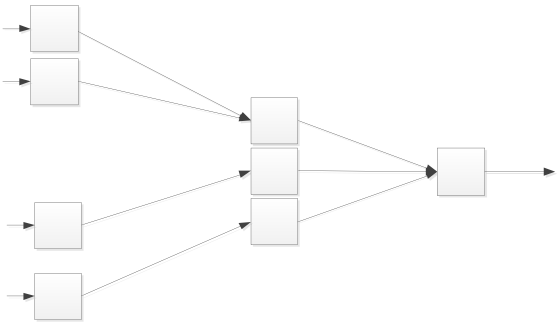

Fig. 8. Formation of user network, representing each address with a node.

In the user network, users are represented as nodes whilst directed edges, which have also amount and timestamp information, represent flow of bitcoins between them. A source node represents a payer, whereas a target node represents a payee. A user node includes Bitcoin addresses (hash of the public key) of the corresponding user. User network cannot be derived from the blockchain directly; it needs extra work. At first, the graph can be constructed by representing each address with a node as shown in Fig. 8. In this figure, each square represents a Bitcoin address, and directed edges are the transfer of bitcoins between addresses.

However, a user may have multiple addresses, since it is suggested to generate a new private-public key pair for each transaction. Reid and Harrigan tried to cluster nodes, which belong to the same user, using this fact. It is not possible to obtain a perfect and true user network, since real owners of Bitcoin addresses are unknown; in this way Bitcoin provides anonymity to some extent. Nevertheless, different Bitcoin addresses that are expected to belong to the same user can be linked by (*i*) utilizing multi-input transactions, (*ii*) utilizing change addresses and (*iii*) behavior-based clustering. Fig. 9 shows clustering of addresses in Fig. 8 into users. Each circle represents a user and contains addresses owned by that user. Directed edges represent transfer of bitcoins between users.

Date interval for each study that analyzes the blockchain data is given in Fig. 6, and some metrics for these studies are given in Table I. The studies of Reid and Harrigan [52], [61], Ron and Shamir [62], [92], Androulaki *et al.* [93], Ober *et al.* [94], Ortega [64], Meiklejohn *et al.* [53], Möser *et al.* [56], Fleder *et al.* [67], Spagnuolo *et al.* [68],

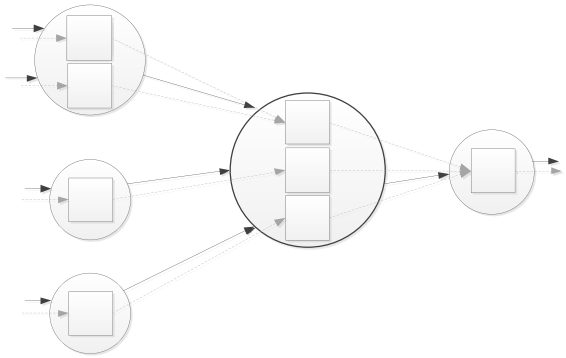

Fig. 9. Formation of user network, clustering addresses to users.

Baumann *et al.* [75], Lischke and Fabian [78], Dupont and Squicciarini [95], Zhao *et al.* [96], [97], Nick [98], Ferrin [99], Yanovich *et al.* [100], and Neudecker and Hartenstein [87] are included in the analyses.

Some additional information for these studies is as follows. Ron and Shamir [92] analyzed transactions related to Dread Pirate Roberts (Ross William Ulbricht) who was running the Silk Road marketplace and was arrested on 1st October 2013. Spagnuolo *et al.* [68] did not state date interval for gathering addresses, according to our calculations using statistics in the Web [101], this number corresponds to the blockchain data up to 22nd April 2013. They also stated they use a transaction graph updated to 1st November 2013 in their scalability experiments. In addition, they investigated activities belonging to two cases related to Dread Pirate Roberts using blockchain data between 30th April 2011 and 2nd July 2011, and between 31st March 2013 and 12th April 2013. Another case study investigates CryptoLocker, between 9th September 2013 and 15th December 2013. Baumann *et al.* [75] and Lischke and Fabian [78] used Bitcoin transaction data from the data scraper tool by Brugere [102]. This tool extends tools developed by Martin Harrigan [103], which extends Gavin Andresen's bitcointools [104].

For all metrics in Table I, the numbers that are provided in the studies are given. However, some metrics are not given in the studies, we calculated estimated values for number of blocks, number of transactions, number of unique Bitcoin addresses by using date intervals of the analysis. We mark these estimated values with a "$\sim$" character. For the studies that do not provide values for the number of blocks metric, we utilized block information in Bitcoin Block Explorer [5]. For the studies that do not provide values for the number of transactions in the analysis metric, we estimated values from the total number of transactions chart [10] in blockchain.info. For the studies that do not provide values for the number of unique Bitcoin addresses in the analysis metric, we utilized cumulative numbers in the number of unique Bitcoin addresses used chart [101]. The metrics are not applicable for two studies [56], [92] since they made analysis including a subset of transactions and they did not provide the numbers.

a) Utilizing multi-input transactions: Different addresses can be linked to a single user utilizing multi-input transactions. A multi-input transaction occurs when a user performs

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE I
METRICS FOR STUDIES THAT ANALYZE ANONYMITY AND PRIVACY IN BITCOIN

Study	Number of blocks in the analysis	Number of transactions in the analysis	Number of unique Bitcoin addresses in the analysis	Grouping of Bitcoin addresses to users
[52, 61]	~135,800	1,019,486	1,253,054	1,253,054 addresses to 881,678 users (with multi-input heuristic)
[93]	140,000	~1,184,551	1,632,648	1,632,648 addresses to 1,069,699 users (with multi-input heuristic), to 693,051 users (with multi-input and change heuristics)
[62]	180,001	~3,141,785	3,730,218	3,730,218 addresses to 2,460,814 users (with multi-input heuristic)
[94]	215,399	~10,798,731	~12,711,115	Not specified
[64]	~226,500	128,240 for multi-input heuristic analysis 274,298 for change heuristic analysis	1,719,312 for multi-input heuristic analysis 383,990 for change heuristic analysis	1,719,312 addresses to 32,956 users with multi-input heuristic 383,990 addresses to 32,261 users with change heuristic
[75]	230,686	~15,894,460	~17,229,680	Not specified
[78]	230,686	~15,894,460	~17,229,680	Not specified
[53]	231,207	16,086,073	12,056,684	12,056,684 addresses to 6,595,564 users with multi-input heuristic 12,056,684 addresses to 3,383,904 users with multi-input and change heuristics
[68]	~232,600	~16,558,905	18,153,279	18,153,279 addresses to 4,077,114 users with multi-input and change heuristics
[95]	~293,350	41,099,115	38,886,789	38,886,789 addresses to 17,472,156 users with multi-input heuristic
[98]	~342,800	~59,230,000	60,880,000	Grouped 70% of the addresses to users with multi-input and change heuristics
[92]	Not applicable	Not applicable	Not applicable	Not applicable
[96, 97]	104,700	34,839,029	35,770,360	35,770,360 addresses to 13,062,822 users with multi-input and change heuristics
[67]	~240	89,806	80,030	80,030 addresses to 54,941 users with multi-input heuristic
[56]	Not applicable	Not applicable	Not applicable	Not applicable
[99]	~350,000	~60,000,000	~112,070,000	Not applicable
[100]	~6,760	~10,000,000	Not applicable	Not applicable
[87]	440,349	172,868,721	~315,684,508	196,963,722 addresses to 88 million users with multi-input heuristic 196,963,722 addresses to 46-72 million users with multi-input and change heuristics

a payment using more than one address by combining these addresses in a transaction. This may happen, for example, when the payment amount is greater than each of the balances in the user's addresses. This fact is also indicated by Nakamoto [1]; multi-input transactions reveal that their inputs are owned by the same user and if the owner of an address that is used in one of these inputs is revealed, then it can be figured out that the other transactions, using other input addresses, belong to the same user. This linking can be simply made by analyzing transactions in the blockchain.

Reid and Harrigan [52], [61] were first to use this heuristic. They formed user network from 1,019,486 transactions among 1,253,054 unique Bitcoin addresses. Before preprocessing step, the network consisted of 1,253,054 nodes, which are actually Bitcoin addresses, and 4,929,950 edges, which are the transactions. After pre-processing step, using the assumption that input addresses of a multi-input transaction belong to the same user, the network is reduced to 881,678 nodes and 1,961,636 directed edges. Most of the studies that analyze blockchain data adopted this heuristic. Ron and Shamir [62] used this heuristic, and they mention two probable types of errors in this approach: (*i*) Some addresses belonging to the same user cannot be detected since there is no evidence; this causes an underestimation error, and the number of this error is expected to be quite a few, and (*ii*) some addresses may be linked incorrectly if several users take part in the same multi-input transaction as inputs; this causes an overestimation error, and this error is not so common. For instance, they

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

tried to merge the addresses of a specific large user using all available transactions, but they managed to identify only about one-quarter of his real addresses. They renamed original transaction graph as the *address graph* and contracted transaction graph as the *entity graph*. In the blockchain data they analyzed, out of 3,730,218 Bitcoin addresses, 3,120,948 of them were senders in at least one transaction, but the remaining 609,270 were just receivers. They used a variant of Union-Find graph algorithm to find sets of addresses which are expected to be owned by the same user and combined 3,120,948 addresses into 1,851,544 different users. After adding remaining 609,270 addresses, they obtained a total of 2,460,814 entities.

Androulaki *et al.* [93] first analyzed the real Bitcoin system. Then, they used a simulator, which simulates the use of Bitcoin in a university as a primary currency and provides ground truth information. They introduced *activity unlinkability* and *profile indistinguishability* as the notions of Bitcoin privacy and defined the metrics, which can be used to quantify them. Ober *et al.* [94] evaluated anonymity in Bitcoin in two properties; anonymity set of Bitcoin and the unlinkability of transactions. It was stated that a Bitcoin user can stay pseudonymous, only in the absence of any external knowledge. For maximizing both the anonymity set of Bitcoin and the unlinkability of transactions; (*i*) having a single address and (*ii*) being active with this address for a short time are suggested. Having a single address increases the anonymity set. Being active with this address for a short time limits transaction linkability. Nonetheless, it is mentioned that this is not achievable in practice and the related factors are discussed. Additionally, they analyzed all transactions occurred between 3rd January 2009 and 6th January 2013, and discovered that the Bitcoin anonymity set is reduced in the last 12 to 18 months due to the entity sizes and the overall pattern of usage becoming more stationary. They also showed that the number of inactive entities and the number of dormant coins reduce the anonymity set. Ortega [64] adopted this heuristic, and he stated that it has 100% chance of being correct and define it as a fact, though it is not. Ortega's analysis resulted in an average of 52.17 Bitcoin addresses per user. He stated that utilizing multi-input transactions is the most reliable method of linking Bitcoin addresses expected to belong to the same user.

Meiklejohn *et al.* [53] noticed that some clusters belong to the same user when they took into account the data that they collected by re-identification attacks and the external data like known (or assumed) addresses, which are found in forums and other Web sites. This is expected, since all addresses may not take part in multi-input transactions. Fleder *et al.* [67] stated that they used a similar technique as Reid and Harrigan [52], [61]. Spagnuolo *et al.* [68] adopted this method as their first heuristic where utilizing change addresses takes part as their second heuristic. Baumann *et al.* [75] and Lischke and Fabian [78] used this method by using blockchain data from the data scraper tool by Brugere [102] which utilizes multi-input transactions. Dupont and Squicciarini [95] and Zhao *et al.* [96], [97] are the other researchers utilizing this heuristic. Nick [98] used ground truth data of 37,585 wallets; he found out that on average 79.76% of Bitcoin addresses belonging to a legacy wallet and 68.59% of Bitcoin addresses belonging to a modern wallet can be learned by linking as the result of this heuristic. Clustering results of all these studies are given in Table I.

Kaminsky [84] showed his acceptance of this heuristic in a sample transaction. He stated that all of the input addresses in a transaction belong to the same user. Möser *et al.* [56] also showed their acceptance of this method in the examples they give for identifying Bitcoin addresses in the transaction graph. They stated that Bitcoin addresses that are referenced as inputs in the same transaction could be interpreted as evidence that they belong to the same user with high certainty. Yanovich *et al.* [100] stated that input addresses of a transaction belong to same user in a non-mixing transaction, although main aim of their study was investigating mixing transactions. They formalized mixing using graph notation and determined four transaction categories according to their relation in a mixing operation. Approximately 10 million Bitcoin transactions were categorized accordingly, and it was discovered that occurrence of mixing transactions is quite often on the blockchain, where mixing transactions constitute about 2.5% of all Bitcoin transactions according to their estimations. Detecting mixing transactions would be useful in applying multi-input heuristic more accurately.

b) Utilizing change addresses: Change addresses are the addresses generated by Bitcoin to allow users to take their changes. If a transaction has two outputs and one of them is an old address, and the other is a new address, then it can be assumed that the new address is the change address and belongs to the user who owns the input address of the transaction, or similar heuristics can be used. Transactions in the blockchain can be analyzed to find out change addresses that are expected to belong to the users who are input to transactions, and these addresses can be linked.

Reid and Harrigan [52] stated that if a transaction is created using a particular client implementation and source code of the client can be accessed, then it can be discovered which was the output and which was the change. Then the address of the change can link to the user who created the transaction. Nevertheless, they only utilized multi-input transactions in their experiments. Kaminsky [84] gave a sample transaction and stated that one of the outputs is likely to be owned by the user who owns the input addresses to that transaction. Although previous studies mention this notion, Androulaki *et al.* [93] were first to use this heuristic. Ortega [64] supposed the output with more decimals in the transaction as the change. He stated that this assumption will not always be true, but it is the most approximate one. When he analyzed the blockchain data, he obtained an average of 11.9 Bitcoin addresses per user.

Meiklejohn *et al.* [53] assumed that a change address has only one input and they looked at the outputs of each transaction. If only one of the outputs complied with this assumption, then they identified that output as the change address. If multiple outputs had one input, which made the change address ambiguous, they avoided labeling a change address for that transaction. They further refined their approach

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

by analyzing real Bitcoin data; if an output of a transaction had received only one input earlier or had been previously used in a self-change transaction (i.e., the input address is used as the change address as well), they chose not to assign a change address for that transaction. With this approach, they obtained 3,383,904 clusters. They were able to identify 2,197 of these clusters which consist of over 1.8 million addresses. This showed the efficiency of this approach compared to the hand-tagging method by the re-identification attacks. Spagnuolo *et al.* [68] used this heuristic in addition to utilizing multi-input transactions. They called the change address as the *shadow address*. They decreased 2,432,834 clusters of users, which were obtained utilizing multi-input transactions, to 2,169,115 and increased ratio of clustered addresses from 60% to 90%. Zhao *et al.* [96], [97] and Nick [98] utilized change address methods together with the multi-input method. Nick introduced two more approaches for detecting changes, where he named them *consumer heuristic* and *optimal change heuristic*. Neudecker and Hartenstein [87] applied these variants of change address heuristic in addition to the multi-input heuristic. Number of users that they obtained ranged from 46 to 72 million according to the variant. Clustering results of all these studies are given in Table I.

Möser *et al.* [56] showed their acceptance of this method in the examples they gave for identifying Bitcoin addresses in the transaction graph. They mentioned that a transaction usually has two outputs, the actual output and the change. This indicates that one of these outputs belongs to the same user who owns the input addresses of that transaction. It is stated that the small output is the change most likely. Although change addresses were mentioned by Ron and Shamir [62] too, they did not mention that this information can be utilized to link addresses. Ferrin [99] remarked that transactional cues can be used to detect change addresses and provided indicators that show which output is likely to be spend-output and which output is likely to be change-output. Yanovich *et al.* [100] stated that one can infer which output address is the change based on the output values.

c) Using behavior-based clustering: Clustering is assigning each object in a set of objects to a group (cluster) such that objects in the same cluster are more similar to each other than to those in other clusters. Behavior-based clustering is clustering by evaluating behaviors of objects. Behavior-based clustering techniques can be used to extract data about the users, like linking Bitcoin addresses that are expected to belong to the same user. Several attributes can be determined and data can be retrieved from the blockchain for the analysis. In addition, by analyzing spending habits of the users, possible information about the physical locations of the users can be determined. By analyzing the times of day at which a user has made transactions, an informed guess can be made as to that user's time zone of residence. Reid and Harrigan [52] mentioned that Bitcoin addresses that are used at similar times over an extended time period may be owned by the same user, therefore clustering can be done using this heuristic. Androulaki *et al.* [93] were first to use behavior-based clustering. In the analysis which they used a simulation, they used behavior-based clustering techniques; k-Means (KMC), and the Hierarchical Agglomerative Clustering (HAC) algorithms. As the attributes, they used time at which transactions took place, indexes of different addresses that appear within transactions (as senders or recipients) and values of BTCs spent by the transaction. Their findings showed that the profiles of 40% of the users can be constructed using behavior-based clustering techniques with 80% accuracy. Ortega [64] stated that transactions made from the same destination at similar times can be from the same user. However, it is mentioned that the certainty of this heuristic is lower than other heuristics. Ron and Shamir [92] used this method to link the Bitcoin addresses that are believed to be owned by Dread Pirate Roberts who ran Silk Road marketplace. They linked sets of accounts where one of the sets is known to belong to him and whose bitcoins are all moved in parallel on the same day from one set to another. As the challenge of such a data mining technique was mentioned to be deciding patterns of behaviors that show a sufficiently strong indicator to make a decision about two accounts belong to the same entity. Dupont and Squicciarini [95] studied on spotting a Bitcoin user's physical location by analyzing user's spending habits. They obtained ground truth data by linking known physical address and Bitcoin address pairs of charities. They collected data from the public user profile pages at bitcointalk. They made an informed guess on time zone of a user by analyzing the times of day at which the user makes transactions. For 518 addresses in their ground truth data, their initial results showed up to 72% accuracy with an average of 11 UTC offsets per guess set, which was effectively more than twice when compared to random guessing.

These described methods can be used in combination to obtain further information.

C. Other Characteristic Properties

Other characteristic properties for the studies that analyze anonymity and privacy in Bitcoin are determined as in Table II, and the relationship of these properties with the methods/outcomes in the taxonomy of studies on anonymity and privacy analysis are given. The studies are analyzed according to these properties and results are given in Table III. In the following, the properties, their relationships with methods and outcomes are described, and the studies having these properties are discussed.

1) Built Bitcoin Client: This property is using a custom built Bitcoin client in the study. This property is valid for the studies *utilizing network* and prefers a modified client according to their needs. The outcome of these studies is *mapping Bitcoin addresses to IP addresses*.

Koshy *et al.* [83] built their own Bitcoin client called CoinSeer since there was not any client that is specialized for data collection and existing clients had integrated functionalities that interfered with their goals. CoinSeer was introduced as a lean tool that is specialized for data collection. Biryukov *et al.* [77] built their own Bitcoin client as well. Their client included functionalities specific for their aim, like sending specific Bitcoin messages upon request or establishing various numbers of parallel connections to the

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE II
RELATIONSHIP OF CHARACTERISTIC PROPERTIES AND METHODS/OUTCOMES

Property	Related Methods	Related Outcomes
Built Bitcoin client	Utilizing Network	Mapping Bitcoin Addresses to IP Addresses
Uses ground truth data	Analyzing Blockchain Data, Utilizing Network	Linking Bitcoin Addresses, Mapping Bitcoin Addresses to Geo-Locations, Mapping Bitcoin Addresses to IP Addresses
Provides metrics to quantify privacy	Analyzing Blockchain Data, Utilizing Network	Linking Bitcoin Addresses, Mapping Bitcoin Addresses to IP Addresses
Proposes privacy enhancing measures	*Applicable to all methods*	*Applicable to all outcomes*
Performs actual deanonymization	Transacting, Utilizing Off-network Information	Discovering Bitcoin Addresses, Discovering Identities
Performs flow analysis	Analyzing Blockchain Data	*Irrelevant to outputs*
Investigates a theft case	Analyzing Blockchain Data, Utilizing Off-network Information	*Irrelevant to outputs*
Gives cost information	*Applicable to all methods*	*Irrelevant to outputs*
Analyzes network using network metrics	Analyzing Blockchain Data, Utilizing Network	*Irrelevant to outputs*
Investigates inactive addresses/bitcoins	Analyzing Blockchain Data	*Irrelevant to outputs*

same Bitcoin server. Biryukov and Pustogarov [90] also used a simple Bitcoin client, which is capable of sending different custom-built Bitcoin messages. They used this client to set address cookies and check previously set cookies.

2) Uses Ground Truth Data: Ground truth data, i.e., actual data, is used in the experiments. Studies *analyzing blockchain data* compared ground truth data with information they obtained from the blockchain data. For these studies, the related outcomes are *linking Bitcoin addresses* and *mapping Bitcoin addresses to geo-locations*, since ground truth data in the studies include Bitcoin addresses belonging to same user and geo-locations related to Bitcoin addresses. Studies *analyzing P2P network* compared ground truth data with information they obtained from the underlying network. For these studies, related outcome is *mapping Bitcoin addresses to IP addresses*.

Ron and Shamir [62] tried to merge the addresses belonging to a specific large user using all available transactions, and then they compared their findings with ground truth data. They found that they could identify only about one-quarter of his real addresses. Androulaki *et al.* [93] implemented a simulator that emulates the functionality of Bitcoin. In this simulator, ground truth information was provided which corresponded to the use of Bitcoin in a university setting. They compared results of the clustering according to their heuristics with the ground truth. Dupont and Squicciarini [95] collected data about the physical location of the owners of certain Bitcoin addresses to use as ground truth data. They used information of charities with known physical locations and published Bitcoin addresses. They also collected data from the public user profile pages at bitcointalk, which share real-world locations and Bitcoin addresses. Nick [98] captured ground truth data about 37,585 Bitcoin wallets and the addresses that they used by exploiting a vulnerability in the implementation of Connection Bloom Filtering [105]. Then he applied address clustering heuristics on these addresses to link Bitcoin addresses that are owned by the same user and evaluated precision and recall using the ground truth data. Fanti and Viswanath [86] utilized timestamp information of nodes receiving transactions as the ground truth data. They compared ordering of the nodes for receiving a transaction with the underlying network connections.

3) Provides Metrics to Quantify Privacy: Metrics that measure privacy in Bitcoin are provided. The studies *analyzing blockchain data* provided metrics related to linking Bitcoin addresses and the studies *utilizing P2P network* provide metrics related to mapping Bitcoin addresses to IP addresses. The outcome of these studies is *linking Bitcoin addresses* and *mapping Bitcoin addresses to IP addresses.*

Androulaki *et al.* [93] introduced two different notions of Bitcoin privacy: *Activity unlinkability* and *profile indistinguishability*. They also provided metrics to quantify these notions. For the activity unlinkability, they focused on address unlinkability, which means not being able to link two different Bitcoin addresses. Profile indistinguishability refers to not being able to reconstruct profiles of all Bitcoin users. Ober *et al.* [94] mentioned two complementary approaches for analyzing anonymity of users; using notions of k-anonymity proposed by Sweeney [106] and unlinkability, which was also used by Androulaki *et al.* [93]. Ober *et al.* stated that using few addresses for a user increases the anonymity set and using an address actively for a short time supports unlinkability, which reduces the probability of identifying a user by using the blockchain data. However, they stated that these are hard to achieve in practice. Möser *et al.* [56] used taint analysis tool of blockchain.info [107] for evaluating anonymity. This tool analyzes Bitcoin addresses, which are used in previous transactions of a transaction to find out the possible origins of the bitcoins and outputs a taint value. Taint value shows the likelihood of a connection between a transaction and an address. This value is as high as the likelihood of the connection. They used this approach to analyze the success of mixing services at anonymization. Feld *et al.* [88] employed k-anonymity metric for evaluating anonymity in autonomous systems.

4) Proposes Privacy Enhancing Measures: Measures that can be taken to improve privacy are given in the study. This property is independent of method or outcome in the taxonomy. In other words, it is applicable to all methods and related to all outcomes.

Reid and Harrigan [52] listed good practices for anonymity such as (*i*) generating a new Bitcoin address for every

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE III
CHARACTERISTIC PROPERTIES FOR STUDIES THAT ANALYZE ANONYMITY AND PRIVACY IN BITCOIN

Property	[52, 61]	[93]	[62]	[94]	[64]	[75]	[78]	[53]	[68]	[95]	[98]	[92]	[96, 97]	[67]	[56]	[83]	[77]	[92]	[99]	[100]	[86]	[87]	[88]
Built Bitcoin client	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	✓	✓	✓	X	X	X	X	X
Uses ground truth data	X	✓	✓	X	X	X	X	X	X	✓	✓	X	X	X	X	X	X	X	X	X	✓	X	X
Provides metrics to quantify privacy	X	✓	X	✓	X	X	X	X	X	X	X	X	X	X	✓	X	X	X	X	X	X	X	✓
Proposes privacy enhancing measures	✓	✓	X	X	✓	X	X	X	X	X	✓	X	X	X	X	X	✓	✓	X	X	X	X	X
Performs actual deanonymization	✓	X	✓	X	X	✓	✓	✓	✓	X	X	X	X	✓	X	X	X	X	X	X	X	X	X
Performs flow analysis	✓	X	✓	X	X	X	X	✓	✓	X	X	✓	✓	✓	✓	X	X	X	✓	X	X	X	X
Investigates a theft case	✓	X	X	X	X	X	X	✓	X	X	X	X	✓	X	X	X	X	X	X	X	X	X	X
Gives cost information	X	X	X	X	X	X	X	X	✓	X	X	X	X	X	✓	✓	✓	✓	X	X	X	✓	X
Analyzes network using network metrics	✓	X	X	X	X	✓	✓	X	X	X	X	X	✓	X	X	X	X	X	X	X	✓	X	X
Investigates inactive addresses/bitcoins	X	X	✓	✓	X	X	X	✓	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

transaction, (*ii*) avoiding to reveal any information related to a Bitcoin address that can be used for identifying, (*iii*) sending varying fractions of Bitcoins to one's own newly generated addresses, and (*iv*) using a trusted mixing service. Nevertheless, they stated that these are not applied in practice universally. In addition, they suggested the use of Bitcoin clients which make the user aware of possible linking between their addresses, within the user interface. For evading utilization of change addresses for identification, Androulaki *et al.* [93] suggested dividing bitcoins according to the required payment amount first and then making the payment without a change. They stated that the use of Multi-Input Multi-Output (MIMO) transactions also make identification difficult by utilizing change addresses.

Biryukov *et al.* [77] proposed mainly two countermeasures for the attack they describe. For preventing TOR blacklisting, they suggested increasing required time or computation for every connection, similar to proof-of-work, to make the attack more costly. Against client deanonymization, they suggested adding random delays after the transactions, which would harden linkability of transactions and distinguishing different clients from the same ISP. However, they stated that adding delay would affect Bitcoin usability negatively. Biryukov and Pustogarov [90] defined a countermeasure against TOR ban, as relaxing the reputation-based DoS protection. This would reduce the effect of DoS attack on the network. Another countermeasure proposed was running a set of "TOR-aware" Bitcoin peers which would regularly download TOR consensus and make sure that Bitcoin DoS countermeasures are not applied to servers from the TOR consensus. This is similar to the approach of Atlas [108], where historical record of TOR exit nodes used to connect to the Bitcoin network is maintained. Maintaining and distributing a safe and stable list of onion addresses was given as another countermeasure, where users which would like to stay anonymous should choose at least one address from this list. There is a similar list [109] which is limited and out of date. For TOR-only users, running two Bitcoin nodes, one over TOR and one without, was suggested in order to make sure about the network properties, by comparing blockchains and unconfirmed transactions. Against fingerprinting attack, adding proof-of-work computation to every address list request was suggested. This would make computationally expensive for an attacker to query each client. Other countermeasures against this attack are described as ignoring address list requests on outbound connections, removing the cached address database file before each session and using only trusted hidden-services. In addition, encrypting and authenticating Bitcoin traffic is stated as an obvious privacy enhancing measure.

Ortega [64] suggested using different wallets for different purposes to avoid linking of addresses. For evading utilization of change addresses for identification, he suggested including some output with a small value, but with a lot of decimals in each transaction. This complicates linking inputs and outputs of a transaction. Nick [98] qualified using a modern wallet as a simple measure for mitigating linking Bitcoin addresses that belong to the same user since a new address is created for receiving a change in modern wallets. He also stated that if

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

wallets would behave similarly, then it will be hard to distinguish transactions that are generated using different wallet implementations. Lastly, he suggested avoiding wallets that use Bloom filters, since they are open to attacks.

5) Performs Actual Deanonymization: Examples of deanonymization of users are provided. *Transacting* and *utilizing off-network information* methods were used for deanonymization in the studies. Related outcomes are *discovering Bitcoin addresses* and *discovering identities*.

Biryukov *et al.* [77] stated that they did not perform a deanonymization attack on real clients for ethical reasons. On the other hand, many papers gave examples of deanonymization. Reid and Harrigan [52], [61] identified the main Slush pool [110] account and the computer hacker group known as LulzSec [111] using off-network data. Meiklejohn *et al.* [53] identified addresses of Mt. Gox by performing re-identification attack. They also demonstrated that an alleged address that belongs to Silk Road truly belonged to Silk Road. Ron and Shamir [62] identified addresses of Mt. Gox, the most popular Bitcoin Exchange site, and Deepbit [112], the largest Bitcoin mining pool. Using the information that is shared by users in Bitcoin forum, Fleder *et al.* [67] uncovered forum users, who transacted with some user that had directly transacted with the Silk Road. They also revealed forum users that transacted with Satoshi Dice [113] and WikiLeaks. Baumann *et al.* [75] identified activities of Mt. Gox using data from blockchain.info. Lischke and Fabian [78] identified the transactions owned by the major entities in the Bitcoin network like Mt. Gox, Satoshi Dice, etc. Spagnuolo *et al.* [68] identified an address of the hitman who was paid by Dread Pirate Roberts, the operator of Silk Road, by querying the blockchain for transactions of 1,670 BTC, which is known to be the amount, on the day that the transaction is known to be performed.

6) Performs Flow Analysis: In a flow analysis, an examination of bitcoin inflows and outflows of a user during a specific period or for a number of transactions, or tracing flows of bitcoins between transactions and specific addresses is done by analyzing blockchain data. Transaction and user networks are used in this type of analysis. This property is irrelevant to the outcomes in the taxonomy.

Reid and Harrigan [52], [61] implemented a tool to observe flow of bitcoins between users over time. They observed the flows in the aftermath of a theft. Meiklejohn *et al.* [53] emphasized the effect of utilizing change addresses on flow analysis. They traced the flows of an address which was alleged to belong Silk Road. Ron and Shamir [62] followed flows for 364 large ($\geq$50,000 BTC) transactions in their user graph and analyzed their relationships. They showed that 348 of these transactions were actual successors of the transaction with 90,000 BTC value, which was recorded on 8th November 2010. Möser *et al.* [56] traced flows for the transactions that they made, using the mixing services they analyzed. Fleder *et al.* [67] traced the transactions up to 7 months before the seizure of Dread Pirate Roberts. Ron and Shamir [92] traced flows between addresses related to Dread Pirate Roberts, starting from the FBI controlled account and going backward. Using BitIodine, Spagnuolo *et al.* [68] discovered an interesting connection between an address that is owned by Dread Pirate Roberts and an address with a balance exceeding 111,114 BTC (more than 22,000,000 USD), likely belonging to the cold wallet of Silk Road although balances of all known addresses of Dread Pirate Roberts were zero. Zhao *et al.* [96], [97] used a breadth-first search on user network to analyze currency flows between Bitcoin addresses. Ferrin [99] examined flow between sample transactions and identified transactional patterns.

7) Investigates a Theft Case: Analysis of a known theft case is done by *analyzing blockchain data* and *utilizing off-network information*. This property utilizes flow analysis and is irrelevant to the outcomes in the taxonomy.

Reid and Harrigan [52], [61] analyzed an alleged theft of 25,000 BTC, which was reported in the Bitcoin Forums [114] by a user named *allinvain.* This amount of bitcoins had a market value of approximately half a million U.S. dollars at the time of the theft. Meiklejohn *et al.* [53] used a list of major Bitcoin thefts [115] and tracked these thefts. They analyzed movements of bitcoins after they were stolen and categorized movements as *aggregation, peeling chain, split* and *folding.* Zhao *et al.* [96], [97] investigated a theft case related to Mt. Gox losing more than 850,000 BTC, which was declared by Mt. Gox on 10th February 2014.

8) Gives Cost Information: Cost for the analysis or the attack is provided in terms of money, storage, or duration. This property is independent of methods, i.e., applicable to all methods. Moreover, it is irrelevant to the outcomes in the taxonomy, since the purpose of this property is not deanonymization.

Biryukov *et al.* [77] estimated the cost of their attack on the full Bitcoin network to be under 1500 EUR per month. Ron and Shamir [92] estimated the cost of their attack to be under 2500 USD per month. Möser *et al.* [56] stated that they spent around 0.08 BTC (8 USD on 12th July 2013) as service and transaction fees. Koshy *et al.* [83] stated that their data collection process required storing 60 GB of data per week. Spagnuolo *et al.* [68] stated that the generation of their database takes approximately 30 minutes on a Quadruple Extra Large High-Memory AWS EC2 instance (26 ECU, 68.4 GB of RAM), and its size is around 15 GB. They also mentioned that it takes approximately 45 minutes to process the whole blockchain using the same machine. Neudecker and Hartenstein [87] stated that their clustering process took about 30 minutes to complete for 440,349 blocks using machines equipped with a Xeon E7-8837 and 512 GB memory.

9) Analyzes Network Using Network Metrics: Bitcoin network is analyzed using network metrics. Transaction and user networks formed by *analyzing blockchain data* can be used in this type of analysis, as well as P2P connection networks obtained by *utilizing P2P network*. However, this property is irrelevant to the outcomes since its purpose is not deanonymization. In general, the aim of the studies having this property is identifying characteristics of Bitcoin and evaluating its evolution.

Reid and Harrigan [52], [61] visualized degree distributions (in- and out-degree) and fitted power-law distributions to these distributions. They also found connected components

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

in the network. They calculated edge number, density and average path length of the transaction and user networks which show the growth and sparsification of the Bitcoin network. Baumann *et al.* [75] constructed the degree distribution for the whole network, which gives an insight into the network usage over time. Degree for every user is calculated by counting and summing ingoing and outgoing transactions (in- and out-degree). Another important metric they used for the network analysis is the average clustering coefficient. Average shortest path and the eigenvector centrality were the other metrics that were calculated for the subgraph containing all transactions equal to and higher than 50,000 BTC. To analyze the structure and the dynamics of the Bitcoin network, Lischke and Fabian [78] used network metrics like degree distribution, clustering coefficient, average shortest path length, and centrality. Zhao *et al.* [96], [97] visualized degree distributions (in- and out-degree) of the user network. Fanti and Viswanath [86] used centrality estimators (timestamp rumor centrality and reporting centrality) in their analyses.

Although Kondor *et al.* [116] formed degree distributions (in- and out-degree) of Bitcoin network and used network metrics as Gini coefficient [117], which is a measure of inequality from economics, we did not include this study in our taxonomy since there are not any analyses from anonymity and privacy aspects. Study of Donet *et al.* [118] is another study that we did not include due to the same reason.

10) Investigates Inactive Addresses/Bitcoins: Addresses that are not actively used and related bitcoins that are not active in circulation are analyzed by *utilizing off-network information* and *analyzing blockchain data*. Ratio of these addresses to all addresses or ratio of bitcoins at these addresses to all bitcoins are calculated. Deanonymization is not related to inactive addresses/bitcoins, so this property is irrelevant to the outcomes in the taxonomy.

Ron and Shamir [62] were first to analyze the addresses that are not used in the input and only used in the output of transactions. They named these addresses as *saving accounts*. They discovered that a significant majority of all bitcoins, over 70% of bitcoins were in saving accounts. Ober *et al.* [94] found out 60% of all bitcoins are not active and called these bitcoins as *dormant*. They stated that the number of dormant bitcoins is an important quantity of anonymity since the addresses including dormant coins reduce the anonymity set because they are not active. They calculated the number of dormant bitcoins as 6.3 million as of January 2013. Meiklejohn *et al.* [53] found the ratio of bitcoins in the addresses that are not used in the input and only used in the output of transactions; they named them as *sink addresses*. They calculated the ratio as 64%, meaning only 4 million bitcoins were in circulation, in their time of blockchain analysis.

IV. Taxonomy of Studies With Anonymity and Privacy Improvements

We classify methods of improving anonymity and privacy in Bitcoin-like digital cash systems that are used in the literature as given in Fig. 10. In this figure, we included a hierarchic numbering to serve as an index. For each method, studies that applied the method is given. Resulting outcomes are given in the bottom of the figure. The outcomes and the methods are explained in detail in the following subsections. The number of the studies that have proposals against network analysis is quite a few compared to the number of the studies that have proposals against network analysis, and a detailed taxonomy cannot be provided for the studies on against network analysis. Therefore, for the sake of readability and clarity, these studies are not included in Fig. 10, although they are described at the beginning of the Methods subsection (Section IV-B).

TABLE IV
Relationship of Outcomes of Analyses and Improvement Methods

Outcome of Analyses	Improvement Methods that Address the Outcomes
Discovering Bitcoin Addresses	Cannot be addressed by the methods
Discovering Identities	Cannot be addressed by the methods
Mapping Bitcoin Addresses to IP Addresses	All methods against network analysis
Linking Bitcoin Addresses	All methods against blockchain analysis except Homomorphic Commitments
Mapping Bitcoin Addresses to Geo-locations	Cannot be addressed by the methods

Outcomes in the taxonomy of studies on anonymity and privacy analysis in Bitcoin, which were described in Section III, and the methods in the taxonomy of studies with anonymity and privacy improvements, which are described in this section, are related to each other as given in Table IV. To be more specific, Table IV includes the improvement methods that address outcomes of the analyses.

Discovering Bitcoin addresses is done by transacting or utilizing off-network information. There is no measure that can be taken against transacting, if the receiver would like to receive bitcoins, then he has to provide his Bitcoin address to the sender. There is no improvement method against utilization of off-network information as well; the measure that can be taken is not sharing any information that will relate identity to Bitcoin addresses in the off-network.

Discovering identities is done by utilizing off-network information, therefore this outcome cannot be addressed by the improvement methods against blockchain analysis or network analysis. Any information that will relate Bitcoin addresses to identity information should not be shared in the off-network to prevent discovery of identities.

Mapping Bitcoin addresses to IP addresses is done by utilizing the network. Therefore, it can be addressed by the methods against network analysis.

Linking Bitcoin addresses is performed by analyzing blockchain data or setting address cookie by utilizing the network. This outcome can be addressed by all methods against blockchain analysis except homomorphic commitments and the methods against network analysis. Homomorphic commitments only hide amount information

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

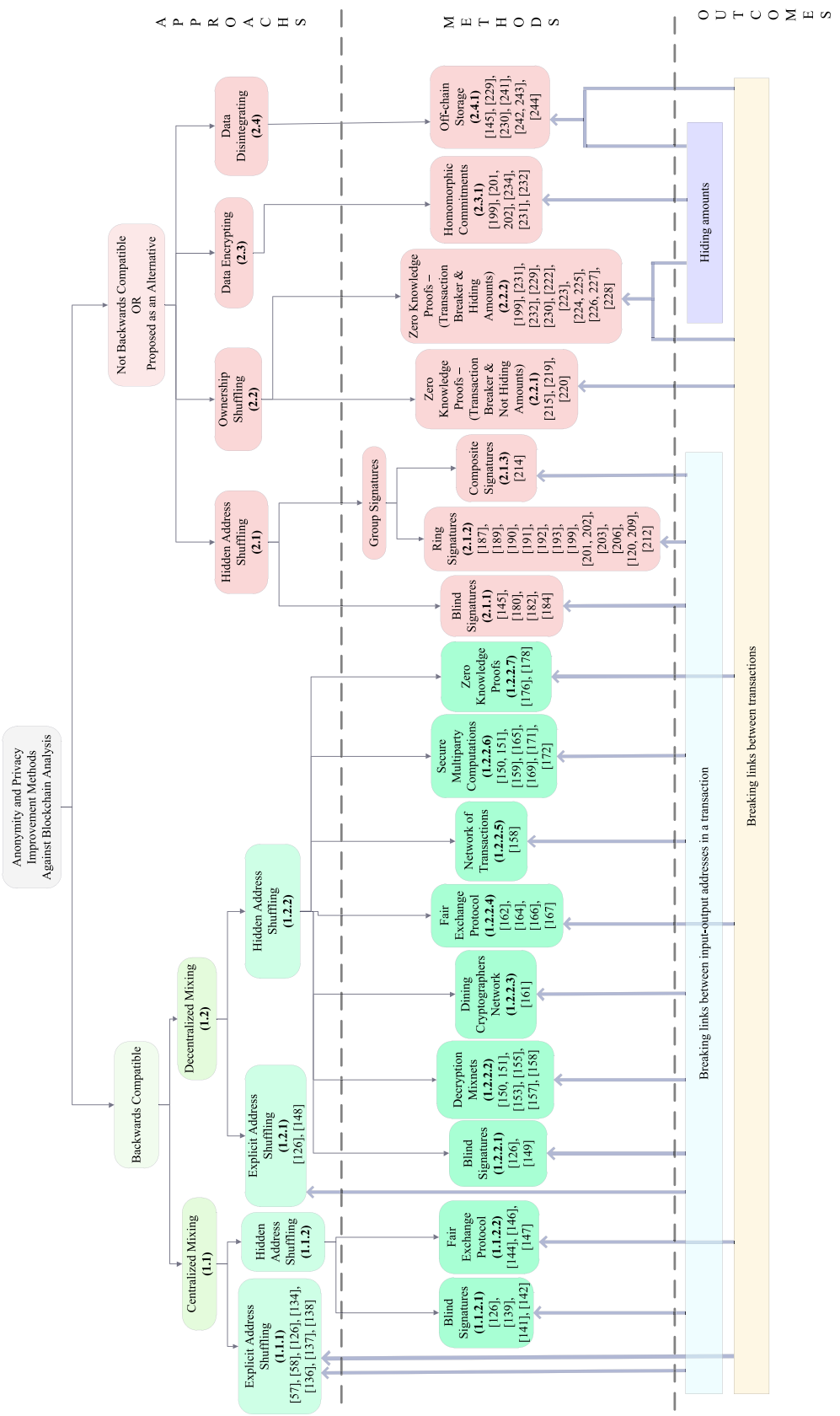

Fig. 10. Taxonomy of anonymity and privacy improvements against blockchain analysis.

in transactions. Therefore, relationships of Bitcoin addresses between transactions remain explicit, and they cannot address this outcome.

Mapping Bitcoin addresses to geo-locations is done by analyzing blockchain data and behavior-based clustering. Times of day that a user makes transactions are used in this type of

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

analysis, and this information cannot be hidden. Therefore, this outcome cannot be addressed by any improvement method.

In the first two subsections of this section, we detail outcomes and methods respectively. In the third subsection, we discuss (i) relationships of proposals with Bitcoin and (ii) performances of the methods.

A. Outcomes

There are four main outcomes of the anonymity and privacy improvement methods. These outcomes are (*i*) breaking links between input-output addresses in a transaction, (*ii*) breaking links between transactions, (*iii*) hiding amounts and (*iv*) hiding IP addresses.

1) Breaking Links Between Input-Output Addresses in a Transaction: Links between inputs and outputs of a transaction are broken, and inputs and outputs cannot be linked. In other words, input-output address links are obfuscated. For an input address in a transaction, one cannot determine the addresses that are output in that transaction, or for an output address in a transaction, one cannot determine the addresses that are input to that transaction.

2) Breaking Links Between Transactions: Two transactions are linked if an output of one of them becomes input to the other. Breaking links between transactions is removing links by adding link obscuring mechanisms in the middle. For a given input of a transaction, the output of another transaction, which becomes the source to that input, cannot be detected if links are broken between two transactions. Similarly, for a given output of a transaction, the input of another transaction, which becomes the destination to that output cannot be detected.

3) Hiding Amount: For improving privacy, amount values in transactions are hidden. However this outcome prevents checking the integrity of the system as a whole, for instance, one cannot count the total number of coins in the system since the amounts are hidden. As a result, if someone can break the system, he can issue coins without being detected.

4) Hiding IP Addresses: IP address of a Bitcoin user is hidden. This results in preventing linking of Bitcoin addresses to IP addresses. This outcome is not included in Fig. 10 since it is the outcome of the methods against network analysis. We did not include the methods against network analysis in the figure either, for the sake of readability and clarity.

B. Methods

Proposals that improve anonymity and privacy in Bitcoin and Bitcoin-like digital cash systems can be divided into two main categories. The first category is the group of proposals against deanonymization by network analysis, and the second category is the group of proposals against deanonymization by blockchain analysis.

The outcome of the methods used for preventing network analysis is hiding IP addresses. Methods against blockchain analysis do not prevent against network analysis, therefore they are suggested to be used in conjunction with the methods against network analysis. The Onion Router (TOR) [35] is the most popular tool used for achieving anonymity, i.e., hiding real IP address, while using the Internet. It is a distributed overlay network [119], comprising TOR nodes designed for providing anonymity for TCP based applications. In TOR, data is encrypted multiple times at the beginning according to the three TOR nodes that the user selects. While traveling in the network, data is routed through a path over that nodes, where decryption of one layer is done at each node, like peeling the layers of an onion, until it reaches its destination. TOR design is based on Chaum's mixnets [32]. It is common among Bitcoin users preferring to use Bitcoin over TOR in order to hide IP addresses. However, Biryukov and Pustogarov discovered that using Bitcoin over TOR leads to new attack vectors [90]. It was stated that an attacker can link a user's transactions, even the ones that are performed using different Bitcoin addresses. The attacker can also control relay of Bitcoin blocks and transactions to the user; he can delay or discard them. Also, users can be fingerprinted while they are using TOR and then they can be recognized when they prefer connecting to the Bitcoin network directly. Stealthcoin [120], which aims to be the primary private currency, has native support for TOR. Anoncoin [121] is another coin that has built-in support of TOR.

TABLE V
COMPARISON OF STUDIES AGAINST NETWORK ANALYSIS

Study	Private currency	Designed for Bitcoin	Encryption type	Encryption scope	Needs modifying Bitcoin protocol	Provides native support for TOR
[35]	X	X	Onion	All blockchain data	X	Not applicable (It is TOR)
[122]	X	X	Garlic	All blockchain data	X	X
[125]	X	✓	Onion	Only new transactions	✓	X
[120]	✓	X	Onion	All blockchain data	X	✓
[121]	✓	X	Onion	All blockchain data	X	✓

I2P (The Invisible Internet Project) [122] is another onion routing tool that creates a hidden network within the Internet. This hidden network is called Darknet. I2P uses "garlic encryption" term instead of "onion encryption" of TOR. As a difference, garlic structure allows adding multiple messages inside the layers of encryption. The main downside of I2P is the low number of outproxies that needed to be used for accessing the regular Internet through I2P. There are Bitcoin clients that are developed to allow running Bitcoin with I2P [123] and Bitcoin exchanges that can be used with I2P [124]. Besides supporting TOR, Anoncoin [121] is designed to be a fully I2P darknet coin.

Transaction Remote Release (TRR) [125] is a new anonymization technology designed for Bitcoin. TRR is inspired by TOR. Its design goal is to defeat attacks that

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

exist while using Bitcoin over TOR. In TRR, routing and multi-layered encryption are like TOR. However, the way of transmitting Bitcoin transactions differs. TOR encrypts all blockchain data. However, TRR only encrypts and transmits new transactions since it is designed specifically for Bitcoin. As a result, the performance and throughput of nodes are improved. The need for modifying Bitcoin protocol is the weakness of TRR. TRR is less vulnerable to man in the middle attacks, but it is vulnerable to DoS attacks.

As a summary, comparison of the studies against network analysis is given in Table V.

Proposals against deanonymization by blockchain analysis can also be examined in two broad categories. The first category is the proposals that are backwards compatible, in which no modification is required to the Bitcoin protocol; thus, the proposed approach can be deployed immediately. The deployment of such a proposal does not affect the soundness of the previous transactions and the blockchain that exist until the deployment. The second category includes the proposals that are not backwards compatible. These proposals are either developed for Bitcoin, but needs modification to the Bitcoin protocol to run, or proposed as an alternative to Bitcoin, i.e., to run independently. These two main categories for improvement proposals are divided into subcategories according to the approaches, protocols, and methods used as shown in Fig. 10 and these are explained in the following subsections.

1) Backwards Compatible: Mixing is the main approach that is adopted by the proposals that are backwards compatible. Mixing can be achieved by obfuscating inputs and outputs of a transaction. Maxwell introduced this idea to the Bitcoin community with his CoinJoin proposal in 2013 [126]. In CoinJoin, which is a transaction formation style to improve privacy, users make joint payments by forming transactions together. Although most of the studies that analyze blockchain data assumed that inputs of a multi-input transaction belong to the same user, Maxwell stressed that it is not a requirement and the opposite is very possible. In CoinJoin, Bitcoin users individually and separately sign a transaction, where they agree on a set of input and a set of output addresses. Then they merge their signatures. As a result, obfuscation is achieved by shuffling the addresses. A transaction formed in this way cannot be distinguished from a transaction that is formed conventionally. Visualization of a sample CoinJoin transaction with three users is given in Fig. 11. Each user provides an input to a transaction, and each user receives an output; however, which output is owned by which user is not known for an outsider, i.e., the ones that do not participate the transaction. The transaction acts as a black box. To increase anonymity, it is important to determine a uniform amount and provide inputs accordingly. This hardens for an outside party to distinguish input and output relations and the anonymity set size becomes the number of parties in the transaction.

CoinJoin can be implemented in both centralized and decentralized ways, as Maxwell described four alternatives. In the first alternative, users can meet over a channel and agree to join in a transaction. In this approach, users learn the input and output addresses of other participants in the transaction. The second alternative is the centralized mixing approach, where

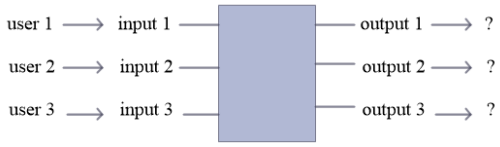

Fig. 11. A sample CoinJoin transaction.

a mixing server gets the requests and performs the mixing. The server learns input and output addresses of the users in this approach. The third alternative is again centralized; however, linking of input and output addresses can be hidden from the mixing server using cryptographic structures like blind signatures, as Maxwell described [127]. The fourth and the final alternative is the decentralized mixing approach, where there is not any third party mixing server and participants in the mixing act as blind signing servers. Although CoinJoin has remained in a forum post, and it was not turned into a paper, it has been widely accepted, and many proposals that are inspired by it came after. There are many implementations of CoinJoin, such as [128]–[130]. It is also included in wallet services such as Darkwallet [131], [132] and Samourai [133].

a) Centralized mixing (1.1): Mixing is performed by a central mixing server. Users that would like to mix their coins share their input and output address information and the mixing server break links between these addresses. A user sends his bitcoins to one of the addresses of the mixing server. These bitcoins are mixed up with other users' bitcoins in the mixing server's pool. Then the user is paid back from another address of the mixing server. In this step, the mixing server uses bitcoins from the mixed pool and gets a mixing fee. Another approach is shuffling a set of user addresses in a single transaction. A centralized mixing service can be implemented in two ways, according to the address shuffling, i.e., relation of input and output addresses, being visible to the mixing service or not.

In the explicit address shuffling (1.1.1), the relation of input and output addresses is explicit to the mixing server. The mixing server can link input and output addresses of the users. Therefore, users cannot stay anonymous against the mixing server, although other parties cannot trace the flow of coins after the mixing. The outcomes of this methods are breaking links between input-output addresses or breaking links between transactions, depending on the number of transactions done to perform mixing. If mixing is done within a single transaction, the outcome becomes breaking links between input-output addresses. Several transactions can also be used to perform mixing, i.e., mixing server can transfer user's assets from user's input address to an address that belongs to the server and then can send to output address of the user from another address that belongs to the server. If several transactions are used, then the outcome becomes breaking links between transactions. CoinJoin [126] can be implemented in this way, such that centralized mixing by explicit address shuffling is performed, as stated before. Bitcoin Fog [57], which was also inspected by Möser *et al.* [56] and found to be successful, can be given as an example of centralized mixing services. Bitcoin Fog does not have any mechanism to

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE VI
COMPARISON OF STUDIES PERFORMING CENTRALIZED MIXING BY EXPLICIT ADDRESS SHUFFLING

Study	Published as a refereed paper	Has implementation
[126]	X	✓
[57]	X	✓
[134]	X	✓
[58]	X	✓
[136]	X	✓
[137]	X	✓
[138]	✓	X

TABLE VII
COMPARISON OF STUDIES UTILIZING BLIND SIGNATURES FOR HIDDEN ADDRESS SHUFFLING IN CENTRALIZED MIXING

Study	Blind signature type	Has implementation	Provided security analysis
[126]	Not provided	No specific given	X
[139]	A blind signature scheme with round-optimal issuing [106]	X	✓
[141]	Elliptic Curve Cryptography (ECC) blind digital signature algorithm	✓	✓
[142]	A blind signature scheme compatible with ECDSA	X	✓

obscure link between input and output addresses from the mixing server. SharedCoin of blockchain.info [134], which was formerly called as Send Shared, was also a centralized mixing service using the approach of CoinJoin. It was one of the mixing services analyzed by Möser *et al.* [56] and found to be successful, as well. Another analysis of SharedCoin was provided [135], which showed that the service provides limited privacy. BitLaundry [58], which was analyzed by Möser *et al.* [56], was another centralized mixing service; however, it was not found successful at mixing; i.e., it allowed linking input and output of transactions. BitLaunder [136] and Coinsplitter [137] are the other examples. Many similar mixing services can be found on the Web, although one should be careful since some of them may be scam. Mixcoin [138], which was published as a refereed paper, is a centralized mixing approach where mixing service gets to know input and output addresses of the users. However, it does not have not any implementation. Mixcoin added an accountability mechanism to mixing against theft with a reputation system, which ensures that a misbehaving mix gets poor reputation. Mixcoin also provides *mix indistinguishability*. Mix indistinguishability is precluding linking of escrow addresses used by mixing services to specific mixing services. Since the input and output addresses are known to mixing server, usage of a series of mixing services was suggested in the study to prevent deanonymization by the mixing server. However, this would increase fees and delays caused by the mixing. Comparison of the studies performing centralized mixing by explicit address shuffling is given in Table VI.

In the hidden address shuffling (1.1.2), the relation of input and output addresses is hidden from the mixing server. The mixing server cannot link input and output addresses. Hidden address shuffling is done using either blind signatures or fair exchange protocols.

- *Blind Signatures (1.1.2.1):* The idea of blind signature came from Chaum in 1983 [3]. In a blind signature scheme, the content of a message is blinded by the message owner using a blinding factor before it is signed. This is required when the signer should not know the content of the message. The signed message can be later verified against the signer's public key. The owner can unblind the message, i.e., remove the blinding factor. Digital cash and electronic voting are the example systems utilizing blind signatures since anonymity is needed. CoinJoin [126] can be implemented using blind signatures, such that centralized mixing by hidden address shuffling is performed, as stated before. Blindcoin [139], which was proposed as an improved version of Mixcoin, hides output address from the mixing server by utilizing a blind signature scheme [140], as a result, the input and output address linking is not explicit to the mixer. The outcome becomes breaking links between input-output addresses in a transaction. In Blindcoin protocol, at a high level, (*i*) the user sends the output address as blinded to the mix, (*ii*) the mix sends a partial warranty back to the user by signing a message with the blinded output, (*iii*) the user transfers bitcoins to the escrow address of the mix, (*iv*) the mix completes the warranty by posting the message to a public log, (*v*) the user unblinds the output address, and finally (*vi*) the mix transfers funds to the output address. Blindcoin does not provide mix indistinguishability property that Mixcoin provides. Another blind mixing scheme based on an Elliptic Curve Cryptography (ECC) blind digital signature algorithm was proposed by ShenTu and Yu [141]. ECC blind signature scheme was preferred over RSA blind signature scheme since its performance was found significantly better. Andreev [142] provided a blind signature scheme compatible with ECDSA to anonymize Bitcoin transactions. Comparison of the studies utilizing blind signatures for hidden address shuffling in centralized mixing is given in Table VII.
- *Fair Exchange Protocol (1.1.2.2):* A fair exchange protocol ensures that either all participating parties get the exchanged item or all of them get nothing [143]. TumbleBit [144] is a scheme that allows anonymous payments through a mixing server, where no trust to the server is required. TumbleBit was built on blindly signed contracts [145], which was not backwards compatible with Bitcoin. It consists of two interleaved fair exchange protocols. In the first, payer swaps bitcoins to an anonymous voucher from the server, and in the

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

second, payee gets bitcoins from the server in exchange for the anonymous voucher. As a result, the link between input and output Bitcoin addresses are broken by performing multiple transactions; therefore, the outcome becomes breaking links between transactions. The fair exchange protocols were implemented using smart contracts via the scripting functionality of Bitcoin and an "RSA evaluation as a service" protocol is at the core of TumbleBit. Another fair exchange protocol that guarantees strong fairness while preserving the anonymity of the consumer and the merchant was provided by Jayasinghe *et al.* [146]. This protocol utilizes an optimistic approach by keeping the role of the trusted third party to a minimum. Finally, Jayasinghe and Jayasinghe [147] proposed another anonymous fair exchange payment protocol which uses an online trusted third party (TTP) to achieve true fair exchange and resolution of conflicts. TTP issues shared symmetric keys to both exchanging parties for secure communication and an anonymous fair exchange transaction is completed in 10 messages between the protocol entities where TTP does not store the transaction content.

b) Decentralized mixing (1.2): No third party, i.e., a central mixing server, is required in the decentralized mixing. Mixing is performed collectively by the participating users. Decentralized mixing can be done by either explicit address shuffling or hidden address shuffling.

In the decentralized mixing with explicit address shuffling (1.2.1), relation of input and output addresses is learned by the participating users; there is not any mechanism to hide. CoinJoin [126] can be implemented in this manner. Cloak coin [148] is another example of decentralized mixing with explicit address shuffling. In Cloak coin, users cloak transactions of other users by providing inputs and outputs and earn a reward in return. An Elliptic Curve Diffie Hellman key exchange (ECDH) is used to derive a shared secret key between the cloaked user and cloaking users. Then, they used symmetric AES-256[2] data encryption to exchange information about inputs and outputs. However, a cloaked user learns input-output address pairs of the cloaking users. Since mixing is done in a single transaction, the outcome is breaking links between input-output addresses in a transaction.

In decentralized mixing with hidden address shuffling (1.2.2), the relation of input and output addresses is hidden from other participating users in the mixing. This can be done using various cryptographic protocols discussed below.

- *Blind Signatures (1.2.2.1):* As mentioned before, Maxwell's CoinJoin [126] can be implemented in a decentralized way by using blind signatures resulting in breaking links between input-output addresses in a transaction. Another study that uses blind signatures focused on secure and joint Bitcoin trading [149], in case of a Bitcoin account is owned by multiple people. Partially blind fuzzy signatures were proposed to ensure anonymity of the multiple owners.
- *Decryption Mixnets (1.2.2.2):* Decryption mixnets were introduced by Chaum [32]. In these structures, a set of inputs pass through a set of mix nodes, where each mix node shuffles the inputs and applies encryption and decryption. CoinParty [150], [151] is a mixing protocol designed using combination of decryption mixnets with threshold signatures [152]. CoinShuffle [153] was proposed as another decentralized mixing protocol which was inspired by CoinJoin and the accountable anonymous group messaging protocol Dissent [154]. In CoinShuffle, users meet through a public bulletin board and agree to participate in a transaction in order to mix their payments. Output addresses in the mixing transaction are shuffled in an oblivious way, as in a decryption mixnet, by using layers of encryption and decryption, resulting in breaking links between input-output addresses. An IND-CCA (INDistinguishability under Chosen Ciphertext Attack) secure public key encryption scheme is required in the protocol. A general approach for pseudonym mixing, based on CoinShuffle, was proposed and a specific design for Bitcoin, which was called BitNym, was given in detail by Florian *et al.* [155]. The main objectives of BitNym are providing unlinkability of pseudonyms to user identities and to other pseudonyms that belong to the same user and providing unlinkable pseudonym changes. A proof of concept implementation of CoinShuffle is given in [156]. Performance analysis of CoinShuffle showed that a small communication overhead induces for a participant and the computation overhead is negligible. Other nodes in the Bitcoin network are affected in minimal in terms of storage and computational costs. Privacy-enhancing overlays [157] is another study that explained how mixing can be achieved by utilizing decryption mixnet approach. In this study, centralized and decentralized mixing approaches were analyzed using taint resistance notion. Methods of grouping users, performing mixing and signing a transaction were described. It is stated that the centralized proposal is similar to Mixcoin and the decentralized proposal is similar to CoinShuffle. Coutu [158] proposed a decentralized synchronous N-to-N mixing model in his master thesis. Within this model, he proposed a new method; private key encryption for three-party mixing. A block cipher that uses a pseudo-random function was used. SecureCoin [159] is another study which is fully compatible with Bitcoin. It uses public key encryption in shuffling of destination addresses. Comparison of the studies utilizing decryption mixnets is given in Table VIII.
- *Dining Cryptographers Network (1.2.2.3):* Dining Cryptographers Network (DC-net) is a method proposed by Chaum [160]. For a DC-net consisting of two users, the users share a key k. When one of the users wishes to anonymously publish a message m, where $|m| = |k|$, he publishes $M_1 = m \oplus k$, where $\oplus$ denotes the *exclusive or* (XOR) operation (bitwise addition modulo 2), and the other user publishes $M_2 = k$. Then the message m can be computed as $M_1 \oplus M_2$ by an observer, however the observer cannot identify the sender. Golle and Juels [89]

[2]Throughout the study, it is stated as "symmetric RSA-256". We made error correction here.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE VIII
COMPARISON OF STUDIES UTILIZING DECRYPTION MIXNETS

Study	Inspired by	Has implementation	Evaluates performance	Has security analysis
[150, 151]	CoinShuffle [153] and Dissent [154]	✓	✓	✓
[153]	CoinJoin [126] and Dissent [154]	✓	✓	✓
[155]	Not specified	✓	✓	✓
[157]	CoinJoin [126]	X	X	✓
[158]	Not specified	X	X	X
[159]	CoinParty [150, 151] and CoinShuffle [153]	X	✓	✓

TABLE IX
COMPARISON OF STUDIES UTILIZING FAIR EXCHANGE PROTOCOLS PERFORMING DECENTRALIZED MIXING BY HIDDEN ADDRESS SHUFFLING

Study	Published as a refereed paper	Method	Provided pairing algorithm	Has implementation	Has security analysis
[162]	✓	A cut and choose protocol [163] and scripting features of Bitcoin	X	X	X
[164]	✓	A cut and choose protocol [163] and scripting features of Bitcoin	✓	X	✓
[166]	X	Hashlocked transactions	X	X	X
[167]	✓	2-of-3 multisignature over Pay to Script Hash (P2SH) scheme	X	X	✓

detail the extension of this protocol to multiple users. DiceMix [161], which was built on the original DC-net protocol, was proposed as a general decentralized mixing protocol, providing sender anonymity, by the authors of CoinShuffle. Moreover, built on CoinJoin and DiceMix, CoinShuffle++ was introduced in the same study. CoinShuffle++ is a decentralized mixing algorithm specifically designed for Bitcoin and it is simpler and more efficient than CoinShuffle. In a simulation for 50 users, a successful transaction was completed in 8 seconds in CoinShuffle++, whereas CoinShuffle required almost 3 minutes. By using dining cryptographers network, links between input-output addresses are broken as the outcome.

- *Fair Exchange Protocol (1.2.2.4):* Two-party decentralized mixing via a fair exchange protocol can be performed using scripting functionalities of Bitcoin. Barber *et al.* proposed a Fair Exchange Protocol [162], which can be used as a two-party mixing protocol. A cut and choose protocol [163], and scripting features of Bitcoin were utilized in the approach which were explained in the study briefly. Pairing, i.e., finding peers to mix with, was not covered in the study. XIM [164] is another two-party mixing protocol which also allows users to find partners to mix with anonymously. Finding mixing partners depends on ads placed on the blockchain. XIM is designed to be a multi-round system to increase security. The fair exchange protocol of Barber *et al.* [162] was utilized in this study; however, the authors state that Secure Multiparty Computations of Andrychowicz *et al.* [165] can also be used. A participation fee was included to prevent DoS attacks. CoinSwap [166] is also a two-party mixing protocol proposed by Maxwell. Fair exchange was achieved by utilizing a special transaction type, called hashlock transactions. CoinSwap transactions look like regular 2 of 2 escrow transactions. The study of Wijaya *et al.* [167] can also be considered as a fair exchange protocol. This protocol requires 5 middlemen in addition to a payer and a payee, totaling 7 participants. 4 groups are formed from these 7 participants, and a series of 2-of-3 multi-signature escrow transactions over Pay to Script Hash (P2SH) scheme are used for the payment and refund. Since several transactions are used in a fair exchange protocol, the outcome is breaking links between transactions. Comparison of the studies utilizing blind signatures for hidden address shuffling in centralized mixing is given in Table IX.
- *Network of Transactions (1.2.2.5):* Coutu introduced the network of transactions approach [158], where a network of transactions consists of a number of small two party switchboxes that are combined in a structured network with the purpose of performing permutation of the addresses. The output of a switchbox is only known the participants of the switchbox since they determine the input and output mapping. Usage of different networking structures was explained in the study, such as random pairing, butterfly network, and omega network. This approach is similar to decryption mixnets, however, encryption and decryption operations are not performed. As the outcome, links between input-output addresses in a transaction are broken.
- *Secure Multiparty Computation (1.2.2.6):* Secure Multiparty Computation (SMC) allows a group of users to compute the value of a public function using their private data, while they keep their inputs private. SMC was introduced by Yao in 1982 [168]. Using SMC for shuffling addresses was first proposed in bitcointalk forum by a member named hashcoin [169]. In this proposal, address shuffling is done using a permutation function in SMC, and links between input-output addresses in a transaction are broken. Andrychowicz *et al.* [165] proposed SMC on Bitcoin based on the coin-tossing protocol of Blum [170]. Although their main purpose

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE X
COMPARISON OF STUDIES UTILIZING SMC

Study	Based on	Published as a refereed paper	Provided security analysis
[169]	No specific given	X	X
[165]	Coin-tossing protocol of Blum [170]	X	X
[171]	No specific given	X	X
[172]	Multi-party sorting [173]	X	✓
[159]	Elliptic curve polynomial/Shamir's secret sharing scheme	✓	✓
[150, 151]	Damgard et al.'s protocol for general Secure Multi-Party Computation [174]	✓	✓

was not privacy or anonymity improvement, it is stated that their approach can be used for fair exchange as well. Rosenfeld [171] stated that n-party mixing transactions can be performed using secure multi-party computation to exchange address information. Yang [172] provided a decentralized, secure multiparty protocol for implementing a mixing protocol using secure multi-sorting [173]. SecureCoin [159] utilized secret sharing schemes and SMC in the first aggregation phase, in which each user deposits to a temporary aggregation address before the address shuffling. The use of a set of mixing peers was proposed in CoinParty [150], [151]. Mixing is performed by using multiple sequential transactions instead of one atomic group transaction to guarantee that mixing is achieved in case of the majority of the users or mixing peers fail or misbehave. CoinParty employs a threshold variant of the Elliptic Curve Digital Signature Algorithm (ECDSA) scheme realized using Damgård *et al.*'s protocol for general SMC protocol [174]. This allows funds to be aggregated via a threshold transaction in commitment phase before address shuffling using decryption mixnets. By evaluating a prototype, scalability of the protocol is shown in the study. Finally, CoinParty was stated to be the first mixing approach for Bitcoin that providing plausible deniability. Comparison of the studies utilizing SMC is given in Table X.

- *Zero knowledge proofs (1.2.2.7):* The concept of zero knowledge protocols was introduced by Goldwasser *et al.* [175]. A zero knowledge proof allows one to prove that a statement is true without giving any other information than the statement is true. Zero Knowledge Contingent Payments (ZKCP) [176] were proposed in 2011 by Maxwell. However, its first usage occurred in 2016 [177]; it was demonstrated by performing a transaction between Maxwell and Sean Bowe, who mostly implemented ZKCP. In this transaction, Maxwell purchased a solution to a 16x16 Sudoku puzzle for 0.10 BTC from Bowe. ZKCP utilizes hashlock transactions and zero knowledge proofs. Another zero knowledge contingent payment protocol was provided by Banasik *et al.* [178], where the authors sought a method of creating non-trivial efficient smart contracts using the standard transactions only. Although the simple cut and choose technique, as provided by Lindell and Pinkas [179], is used in the approach for efficiency, zero knowledge protocols are used in order to describe how the protocol can be generalized. In zero knowledge protocols, mixing is achieved in multiple transactions, therefore links between transactions are broken.

2) Not Backwards Compatible / Proposed as an Alternative: There exist several improvement proposals that require modification, some of them are designed to be used with Bitcoin, whereas some of them are inspired by Bitcoin and designed to be similar to Bitcoin but completely independent, proposed as an alternative Bitcoin-like digital cash system.

a) Hidden address shuffling (2.1): In the hidden address shuffling, sender and/or receiver Bitcoin address(es) is/are shuffled with other Bitcoin addresses. Moreover, which output address corresponds to which input address remains hidden. The aim is to break traceability of bitcoin flows. Methods in this approach are using blind signatures, ring signatures and composite signatures.

- *Blind Signatures (2.1.1):* Ladd [180] introduced a new method of forming transactions where blind signatures are used with cut and choose. Modification to the scripting functionalities of Bitcoin, like the addition of a new opcode and a new signature type is required. Blindly Signed Contracts [145], which is the study done prior to TumbleBit, uses blind signatures and smart contracts to implement a fair exchange protocol. They used Boldyreva's scheme [181], instantiated with elliptic curves for which the Weil or Tate pairing are efficiently computable, and the computational Diffie-Hellman problem is sufficiently hard. Although Bitcoin supports elliptic curve operations, the curve that is used does not support the required bilinear pairings. Therefore, there is a need for adding an opcode that supports elliptic curves with efficient bilinear pairings; thus, modification is required. Darkcoin [182], which is a privacy-centric cryptographic currency based on Bitcoin, uses a decentralized implementation of CoinJoin, which is called DarkSend. In DarkSend, transactions are merged together into a larger anonymous transaction, resulting breaking links between inputs and outputs addresses in the transaction. Inputs of the same size, which are called denominations, are allowed for participating into the DarkSend pools for improving anonymity. An ECC-based blind signature scheme [183] is used for proving the provided outputs belong to one of the participant users of the pool. Master node structure was introduced, where a master node is elected to create the transaction in a decentralized fashion. Darkcoin was later turned into Dash [184], [185], which is 8th in crypto-currency market capitalization list, having $11.4 billion market cap as of December 2017 [8]. In Dash, a chaining approach is adopted to increase anonymity. In this approach, funds are

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE XI
COMPARISON OF STUDIES UTILIZING BLIND SIGNATURES – NOT BACKWARDS COMPATIBLE

Study	Signature type	Required modifications to Bitcoin protocol
[180]	Blind signatures with DSA/ECDSA, modified to include cut-and-choose	Addition of a new opcode and a new signature type
[145]	Boldyreva's [181] scheme	Addition of an opcode that supports elliptic curves with efficient bilinear pairings
[182]	An ECC-based blind signature scheme [183]	Master node structure
[184]	An ECC-based blind signature scheme [183]	Chained master node structure

sent through a series of Master nodes. Comparison of the studies utilizing blind signatures that are not backwards compatible is given in Table XI.

- *Ring Signatures (2.1.2):* A ring signature is a special type of a group signature, where there is not any group manager. This signature type was introduced by Rivest, Shamir, and Tauman in 2001 [186]. Any member of the group can sign using the ring signature, and the signing member cannot be identified by the ring signature. CryptoNote [187] was proposed as a base scheme that includes solutions to the main deficiencies of Bitcoin and CryptoNote can be utilized as a framework by other digital cash mechanisms. It uses one-time ring signature which is a type of group signature and based on traceable ring signature of Fujisaki and Suzuki [188]. In CryptoNote, for sender privacy, the address of the sender is grouped with other addresses using ring signatures; the sender produces a signature (one-time ring signatures with non-interactive zero-knowledge proofs) which can be verified using a group of public keys, not a single public key. This makes impossible for one to tell which address is the actual address of the sender. For receiver privacy, users publish a single address, and the destination of each CryptoNote output is a new address, which is derived from recipient's address and sender's random data. Therefore, every destination address is unique by default and address reuse does not occur by design, which makes the linking of addresses impractical. As the result, links are broken between input and output addresses in a transaction. Diffie-Hellman key exchange protocol is used. Bytecoin [189] is the first cryptocurrency that used CryptoNote as a base. DigitalNote [190] uses CryptoNote code base for unlinkable transactions and a protocol for transferring encrypted messages within transactions is added where a symmetric stream cipher is used. DarknetCoin [191] and Aeon [192] are other cryptocurrencies that were implemented using the CryptoNote framework. Monero [193] was introduced as a privacy-focused cryptocurrency, which has $6.5 billion market cap and is the 11th cryptocurrency in the crypto-currency market capitalization list as of December 2017 [8]. The original Monero protocol was not based on Bitcoin's code; it was initially based on CryptoNote protocol and Bytecoin reference code. As CryptoNote, Monero used one-time ring signatures to hide the source and destination of the transactions, and a detailed description was provided by Noether and Noether [194]. Nevertheless, CryptoNote protocol was analyzed in detail by the Monero team, and the results were provided in several reports. A review of CryptoNote whitepaper was provided by Noether [195]. The problems with the protocol and deficiencies were given in the review. Investigation results on traceability and security in CryptoNote under attacks were also presented by Noethers, Mackenzie and Monero Core Team [196], [197]. Another research bulletin of Monero Research Lab [198] analyzed the attack, which was executed against the Monero cryptocurrency network on 4th September 2014. The attack partitioned the network into two distinct subsets, which refused to accept the legitimacy of the other subset. The attack was found to be due to the deficiencies in the CryptoNote protocol and fixes were provided in the study. One of the cryptographic structures that Maxwell's Confidential Transactions [199] use is Borromean ring signatures [200]. To further improve the privacy provided by Monero, hiding amounts of transactions using a new type of ring signature; multi-layered linkable spontaneous anonymous group signature was proposed. This proposal leveraged Maxwell's approach of Confidential Transactions, which was combined with ring signatures, resulting Ring Confidential Transactions [201], [202] hiding sender and receiver, as well as hiding amount information. Boolberry [203] is another cryptocurrency based on CryptoNote. The Boolberry team found out that the ring signatures take up most of the transaction size in CryptoNote. However, it was stated that the ring signature is not needed for an old transaction that got a lot of confirmations, even if the transaction's output is not spent yet. Therefore, ring signatures are cut off from old transactions in Boolberry, resulting Boolberry blockchain to be at least 55% and up to 90% smaller compared to CryptoNote [204]. The main anonymity improvement of Boolberry compared to CryptoNote is adding a special flag in each transaction's output [205]. This ensures that the output cannot be used without mixing with other addresses. ShadowCash [206]–[208] was proposed as an anonymous cryptographic transaction protocol. In ShadowCash, traceable ring signatures, which utilize a non-interactive zero knowledge proof, are used. Nevertheless, ShadowCash does not require a trusted setup as in ZeroCoin and ZeroCash. Stealthcoin [120], [209] proposed using Chandran signatures [210], which is a special type of ring signatures. A Chandran signature has a sub-linear size, where a regular ring signature has a linear size. Usage of Franklin and Zhang's unique ring signature protocol [211] was examined by Mercer [212] to improve privacy in Bitcoin. In unique ring signatures, tags are used to link signatures that have same signer, message, and ring. It was stated that the privacy of the scheme relies on

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE XII
COMPARISON OF STUDIES UTILIZING RING SIGNATURES

Study	Ring signature type	Based on CryptoNote [187]	Hides amount
[187]	One time ring signature based on traceable ring signature of Fujisaki and Suzuki [188]	Not applicable (It is CryptoNote)	X
[189]	One time ring signature based on traceable ring signature of Fujisaki and Suzuki [188]	✓	X
[190]	One time ring signature based on traceable ring signature of Fujisaki and Suzuki [188]	✓	X
[191]	One time ring signature based on traceable ring signature of Fujisaki and Suzuki [188]	✓	X
[192]	One time ring signature based on traceable ring signature of Fujisaki and Suzuki [188]	✓	X
[193]	One time ring signature based on traceable ring signature of Fujisaki and Suzuki [188]	✓	X
[199]	Borromean ring signature [200]	X	✓
[201, 202]	Multi-layered linkable spontaneous anonymous group signature	✓	✓
[203]	One time ring signature based on traceable ring signature of Fujisaki and Suzuki [188]	✓	X
[206-208]	Traceable ring signatures	X	X
[120, 209]	Chandran signatures [210]	X	X
[212]	Franklin and Zhang's unique ring signature [211]	X	X

EC-DDH (Elliptic Curve Decision Diffie-Hellman) and the scheme is secure under the EC-DDH assumption in the random oracle model. However, many open problems and requirement of further improvements and research were mentioned in the study. Comparison of the studies utilizing ring signatures is given in Table XII.

- *Composite Signatures (2.1.3):* Composite signatures are extension of aggregate signatures [213]. A composite signature combines a number of individual signatures, where there is not any order among them. It allows adding more signatures at any time and it is computationally hard to obtain individual signatures from the composite signature. Saxena *et al.* [214] used composite signatures to enhance anonymity in Bitcoin-like cryptocurrencies. The links between inputs and outputs are removed in a transaction with a slight modification to the protocol. Multiple transactions are combined into a larger transaction using composite signatures. Individual signatures cannot be obtained from a composite signature; therefore, input and output address linking is obfuscated, and the approach provides plausible deniability. Construction of composite signatures uses bilinear pairings on elliptic curves. Although the approach is similar to CoinJoin, it does not require interaction with other users, and inputs and outputs of other users participating in the transaction need not be known beforehand as in CoinJoin.

b) Ownership shuffling (2.2): The ownerships of coins are shuffled in ownership shuffling approach. This is achieved by breaking coin and ownership connection, at the same time storing which user owns how many coins. Then, a user can prove that he owns a certain amount of coins and spend them. In this way, the ownerships of the coins are shuffled, and coins cannot be tied to users. Ownership shuffling can be achieved utilizing zero knowledge proofs.

- *Zero knowledge proofs (Transaction breaker & Not hiding amounts) (2.2.1):* Zerocoin [215], which was one of the first proposals for improving anonymity in Bitcoin, is a cryptographic extension to Bitcoin and utilizes zero knowledge proofs. It breaks links between transactions without adding trusted parties. In Zerocoin, bitcoins can be converted to zerocoins and then spending any zerocoin can be achieved by showing the validity of a zerocoin by proving that it belongs to a public list of valid coins. The zero knowledge proofs that are used are instantiated using the technique of Schnorr [216], then these proofs are converted into non-interactive proofs by applying the Fiat-Shamir heuristic [217]. Rather than using an expensive OR proof, they used an accumulator based on the strong RSA assumption [218]. However, a double-discrete logarithm proof causes to large proof sizes and verification times. One year later, Garman *et al.* [219] Zerocoin, provided extensions to Zerocoin. They decreased the size of the proofs and removed the random oracle assumption of Zerocoin for the zero knowledge property of the proofs. Pinocchio Coin [220] was proposed as a variant of Zerocoin and suggested using elliptic curves and bilinear pairings to accomplish what Zerocoin has done. Pinocchio [221], which is a pairing-based proof system, was utilized in Pinocchio Coin. However, the proposal is a preliminary case study, and there is not any implementation. Moreover, Zerocoin and Pinocchio Coin require conducting a one-time setup of the parameters of the system by a trusted party. Although CryptoNote [187], DigitalNote [190] and ShadowCash [206] utilized zero knowledge proofs for constructing signatures, their main tools for providing anonymity are ring signatures, and therefore they are not included in this class.
- *Zero knowledge proofs (Transaction breaker & Hiding amounts) (2.2.2):* EZC [222], an extension of Zerocoin, was proposed to hide transaction amounts and address balances which Zerocoin cannot achieve since Zerocoin requires zerocoins to be converted back to bitcoins in order to spend them. EZC achieves this by allowing construction of multi-valued zerocoins with values only known the parties in a transaction and spending zerocoins without converting them back to bitcoins. Similarly to Zerocoin, EZC used accumulators and zero knowledge proofs of knowledge protocols. In addition, EZC improves communication overhead incurred in Zerocoin. Zerocoin was turned into Zerocash protocol [223], which is more efficient than Zerocoin. In Zerocash,

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE XIII
COMPARISON OF STUDIES UTILIZING ZERO KNOWLEDGE PROOFS – TRANSACTION BREAKER & HIDING AMOUNTS

Study	Zero Knowledge type	Built on
[222]	Non-interactive proofs by applying the Fiat-Shamir [217] heuristic to Schnorr [216] and its extensions	ZeroCoin [215]
[223]	Zero Knowledge Succinct Non-interactive ARguments of Knowledge (ZK-SNARKs) [22]	Not specified
[224, 225]	Zero Knowledge Succinct Non-interactive ARguments of Knowledge (ZK-SNARKs) [22]	ZeroCash [223]
[226, 227]	Zero Knowledge Succinct Non-interactive ARguments of Knowledge (ZK-SNARKs) [22]	ZeroCash [223]
[228]	Zero Knowledge Succinct Non-interactive ARguments of Knowledge (ZK-SNARKs) [22]	ZeroCash [223]
[229]	Zero Knowledge Succinct Non-interactive ARguments of Knowledge (ZK-SNARKs) [22]	Not specified
[230]	Non-Interactive Zero-Knowlege Proofs (NIZKP)	Not specified
[199]	No details provided	Not specified
[231]	No details provided	Confidential Transactions (CT) [199]
[232]	Non-Interactive Zero-Knowledge Proof (NIZKP)	Not specified

ZK-SNARKs [23] are used to hide inputs, outputs and amount information of a transaction. Zerocash requires conducting a one-time setup of the parameters of the system by a trusted party as Zerocoin. Zcash [224], [225] is the full-fledged ledger based digital currency which is the implementation of Zerocash protocol. Zcash has a $2 billion market cap as of December 2017, and it is the 17th crypto currency listed in crypto-currency market capitalizations [8]. Komodo [226], [227], which has a $9,330,187 market cap and 27th cryptocurrency in the same list, and Ebitz [228] are new protocols introduced in late 2016. Komodo and Ebitz use zero knowledge proofs of Zcash for hiding sender, receiver and amount information. CoinWitness [229] is another proposal by Maxwell that used ZK-SNARKs to construct compact proofs. Hawk [230] was proposed as a decentralized smart contract system. It provides a framework for building privacy preserving smart contracts. In Hawk, a Zerocash-like protocol is adopted for implementing private cash transfer, therefore zero knowledge proofs are utilized. Confidential transactions [199] are also built using zero knowledge proofs, therefore Bitshares [231] also utilizes zero knowledge proofs, since it uses confidential transactions. In compact confidential transactions [232], a short Non-Interactive Zero-Knowledge Proof is used for showing that for each output that the sum does not overflow. Comparison of the studies utilizing zero knowledge proofs that are transaction breaker and hiding amounts is given in Table XIII.

c) Data encrypting (2.3): Privacy is preserved by encrypting data in this approach. Homomorphic commitments are used for encryption.

TABLE XIV
COMPARISON OF STUDIES UTILIZING HOMOMORPHIC COMMITMENTS

Study	Method for hiding amounts	Uses CT [199]
[234]	ZKP by Schoenmakers and EC-Schnorr signatures	X
[199]	Additively homomorphic commitments (Pedersen)	Not applicable (it is CT)
[231]	Additively homomorphic commitments (Pedersen)	✓
[232]	Elliptic point commitments [198] and Non-Interactive Zero-Knowledge Proof (NIZKP)	X
[201, 202]	Additively homomorphic commitments (Pedersen)	✓

- *Homomorphic Commitments (2.3.1):* Homomorphic commitments allow committing a value without revealing it to the other parties by utilizing homomorphic encryption technique. A homomorphic encryption scheme allows performing computations on ciphertext where the decryption of result gives a value that is equal to the result of operations performed on plain text [233]. This technique was first proposed by Back [234]. He proposed using ZKP by Schoenmakers and EC-Schnorr signatures for hiding amount information of transactions, namely makes amounts encrypted and visible to only to the participants in a transaction. Confidential Transactions (CT) was proposed by Maxwell [199]. The approach utilized cryptographic technique of additively homomorphic commitments for Bitcoin, inspired by Back's proposal [234]. Pedersen commitments are the basic structure that CT is based on. Pedersen commitments can be added, and the sum of the commitments equals to the commitment of the sum of the data. Confidential transactions are used in Elements project [235] by Blockstream [236]. Elements project includes the usage of side chains, which are extensions to existing blockchains, in order to add new features like smart contracts and confidential transactions in order to improve privacy and functionality [237]. Bitshares [238] was introduced as a peer to peer polymorphic digital asset exchange with its first whitepaper in 2013. It was stated that implementing Zerocoin [215] or a similar protocol for user privacy was evaluated in this whitepaper. Also, it was mentioned that the communication between the peers was designed to be encrypted. Two years later, in 2015, two more whitepapers of Bitshares 2.0 were published [231], [239] and it was stated that confidential transactions were used to improve anonymity [231]. Lukianov proposed a compact version of confidential transactions [232] using elliptic point commitments [240]. As in Maxwell's original proposal, homomorphic commitments are used to hide transaction amounts and ensure that the sum of the transaction inputs matches the sum of its outputs. The required commitments are an order of magnitude smaller than the ones that are needed for Maxwell's Confidential

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

Transactions and do not need usage of ring signatures. Ring Confidential Transactions [201], [202] are the combination of Maxwell's approach of Confidential Transactions with ring signatures. Ring Confidential Transactions are included in Monero [193], resulting combination of outputs of both methods, i.e., hiding amounts and input-output address links. As a summary, comparison of the studies utilizing homomorphic commitments is given in Table XIV.

d) Data disintegrating (2.4): Data is disintegrated and stored in blockchain partially in this approach. Data that is not stored in blockchain remains off-chain. For instance, the blockchain may store some transactions, and remaining transactions may stay between only sender and receiver. Another example is using blockchain for storing only hash of the transactions. Thus, we name the method used in this approach as off-chain storage. Data that will be kept off-chain is up to the design and the additional methods used.

- *Off-chain storage (2.4.1):* In this approach, all transaction data are not stored in blockchain, but some data are stored off-chain. Utilizing off-chain storage results in improved scalability, broken links between transactions and hidden amounts. Maxwell's CoinWitness [229] proposed using Zero Knowledge Succinct Non-interactive ARguments of Knowledge (ZK-SNARKs) [22] to construct proofs of correctness which shows a side chain payment is valid. However, it was stated that the method depends on new cryptographic techniques which may suffer from security weaknesses and more study is needed for improving performance. In Hawk [230], transactions are not stored with full financial data in the blockchain. Cash flows and transaction amounts are hidden in the private contracts; therefore, hidden from the public view. In another proposal, tonych [241], a user in bitcointalk forum, showed that all transaction data can be hidden by just using hashing, and no other cryptographic structures. In this approach, only hash of inputs and outputs are stored in blockchain. Plaintext of inputs and outputs are not published in the network and only sent to the receiver, who makes the all validation work. A method of exchanging private transaction data is required. Receiver calculates the hash and verifies it from the blockchain. For double-spending protection, a spending proof is required to be published for each output being spent. Spending proof is the hash of the output being spent. If a user tries to spend an output that is previously spent, this is detected since the calculated spending proofs are the same. In case of such a situation, receiver rejects the transaction. All transaction history of an output being spent has to be received by the new owner (receiver), needs to be stored and passed to the next owner. This requirement results in rapid growth of ownership history, which causes scalability issues and some limitations are proposed to prevent this. One example of the limitations is requiring each transaction to have a single input. Byteball [242], [243] is the first coin that implemented this approach. There is not any block structure in Byteball and transactions are linked in a DAG (Directed Acyclic Graph) structure. Assets can be public or private. Transactions with public assets are published to the chain and become visible to all users. On the other hand, transactions with private assets are not published and only hash values are stored. Another off-chain scheme, which uses micropayment channel networks was provided by Heilman *et al.* [145]. In this scheme, some transactions are not recorded in blockchain and stay only between sender and receiver upon establishing a pairwise micropayment channel after forming an escrow transaction. Lastly, Quantum Bitcoin [244] is a proposal of a Bitcoin-like currency that runs on a quantum computer and based on no-cloning theorem [245] of quantum mechanics. In Quantum Bitcoin, local transactions that are only between sender and receiver are used and no-cloning theorem acts as a copy-protection mechanism to prevent double-spending. For the realization of the proposal, quantum computers are needed to be available, which are also expected to be a threat for the cryptography behind Bitcoin [246]. Comparison of the studies utilizing off-chain storage is given in Table XV.

TABLE XV
COMPARISON OF STUDIES UTILIZING OFF-CHAIN STORAGE

Study	Method	Stored off-chain (hidden from public)
[229]	ZK-SNARKs and hashing	All transaction data in side chain payments
[230]	Private smart contracts	Transaction amounts and cash flows
[241]	Hashing (storing only the hash of inputs and outputs in the blockchain)	All transaction data
[242, 243]	Hashing (storing only the hash of inputs and outputs in the blockchain)	All transaction data
[145]	Micropayment channel networks	All transactions between a pair except an escrow transaction
[244]	No-cloning theorem [245]	All transaction data

C. Discussion

In this subsection, first, we show the relationships of anonymity and privacy improving proposals with Bitcoin. Then we present the performance comparison of the methods.

1) Relationships of Proposals With Bitcoin: We show relationships between Bitcoin and new proposals enhancing anonymity and privacy in Fig. 12. In this figure, we use representation of $X \rightarrow Y$ and we classify the relationships as (i) extension, (ii) modification, and (iii) used. Definitions of each relationship class are as follows:

- *extension:* Y is an extension of X and Y is backwards compatible with X. We use this category for showing the proposals that are extensions to Bitcoin and backwards compatible with Bitcoin.
- *modification:* Y is a modified version of X and Y is not backwards compatible with X. The modification is related to the method of improving anonymity and privacy.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE XVI
PERFORMANCES OF THE METHODS

Method	Slow	Acceptable	Fast
Explicit Address Shuffling (CoinJoin-like)			✓
Blind Signatures		✓	
Fair Exchange Protocol			✓
Decryption Mixnets		✓	
Dining Cryptographers Network			✓
Network of Transactions			✓
SMC		✓	
Zero Knowledge Proofs	✓		
Ring Signatures		✓	
Composite Signatures		✓	
Homomorphic Commitments	✓		
Off-chain storage using ZK-SNARKs	✓		
Off-chain storage not using ZK-SNARKs			✓

- *used: X* used *Y* as it is. In other words, *Y* does not include any change related to the method of improving anonymity and privacy.

According to our observations, all blockchain based digital cash systems are inspired by or derived from Bitcoin, therefore Bitcoin takes part at the center of the figure. We can list CoinJoin, CryptoNote, Zerocoin and Confidential Transactions as the most utilized proposals by the other studies.

2) Performance Comparison of the Methods: While evaluating anonymity and privacy enhancing techniques, efficiencies of the methods should also be considered for applicability. We list general performances of the methods based on our research in Table XVI. Methods listed as *slow* include time-consuming cryptographic structures and stated as slow explicitly in several studies. *Fast* methods either do not require cryptographic structures or include mechanisms that are found to be fast, e.g., hashing or symmetric encryption, by the studies. Methods *acceptable* in performance are in between, i.e., studies do not mention them as exact slow or exact fast. In general, methods requiring asymmetric cryptography operations are classified as acceptable.

Explicit address shuffling, fair exchange protocols, Dining Cryptographers network, network of transactions and off-chain storage methods, which do not use ZK-SNARKs, are the fastest methods since they do not include heavy cryptographic structures. Blind signatures, decryption mixnets, SMC, ring signatures, composite signatures have acceptable efficiencies. Homomorphic commitments, zero knowledge proofs and off-chain storage methods that are using ZK-SNARKs are the slowest methods due to the heavy cryptographic structures that they use.

Justifications for these classifications are given below. Explicit address shuffling does not include any cryptographic structure, therefore it is one of the fastest methods. Blind signatures include asymmetric key operations. Alternatives especially utilizing ECC are found to be efficient [247], [248]. Thus, we classify blind signatures as acceptable. Fair exchange protocols require several transactions for exchanges of the coins. They do not need heavy cryptographic algorithms, in general, they use hash functions. Thus, their performance is classified as fast, as also shown by [144] and [249]. In decryption mixnets, several encryption and decryption operations are performed. However, the feasibility of the method is shown by the studies evaluating performance [150], [156], [159]. Therefore, we classify this method as acceptable. Dining Cryptographers Network [160] requires just symmetric key operations, therefore we classify it as fast. In the network of transactions method [158], mixing is achieved by using a network of two-party mixing switchboxes. Since there is no encryption and decryption, we classify this method as fast.

The computational complexity of SMC was found to be high. However, studies in the recent years [250], [251] show that efficiency of SMC is improved. As a result, we classify performance of SMC as acceptable. Zero knowledge proofs are classified as slow. One of the first studies using zero knowledge proofs is Zerocoin [215], and the authors stated that it requires significant computational effort since double-discrete-logarithm proofs of knowledge are used. Although researchers continue studying on the efficiency of zero knowledge proofs, software implementations of ZK-SNARKs are still found to be immature and slow [252] as of December 2017. Moreover, Zcash authors [253] show the slowness of the current Zcash performance and give information about their recent significant performance improvement studies for the ZK-SNARKs. Although more practical ring signature schemes have been sought [211], [254], we could not detect any research qualifying ring signatures as slow or fast. Our impression is that ring signatures have reasonable efficiency since asymmetric key operations are performed. Noether [201] stated the efficiency as well. Therefore, we classify performance of the ring signatures as acceptable.

Composite signatures are found to be efficient [214], resulting us to classify them as acceptable. Performance problems and need of improvement of the efficiency of homomorphic commitments are stated in several studies [255]–[257]. Thus, we classify them as slow. Off-chain storage methods utilizing ZK-SNARKs are also classified as slow due to the slowness of the ZK-SNARKs, although they exploit the advantages of off-chain storing. Off-chain storage not using ZK-SNARKs utilize micropayment channels or simple cryptography as hash functions. The efficiency of micropayment channels is shown in [258] and [259]. Therefore, we classify them as fast.

V. SUMMARY AND LESSONS LEARNED

After we review the literature related to the anonymity and privacy in Bitcoin and similar digital cash systems, in this

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

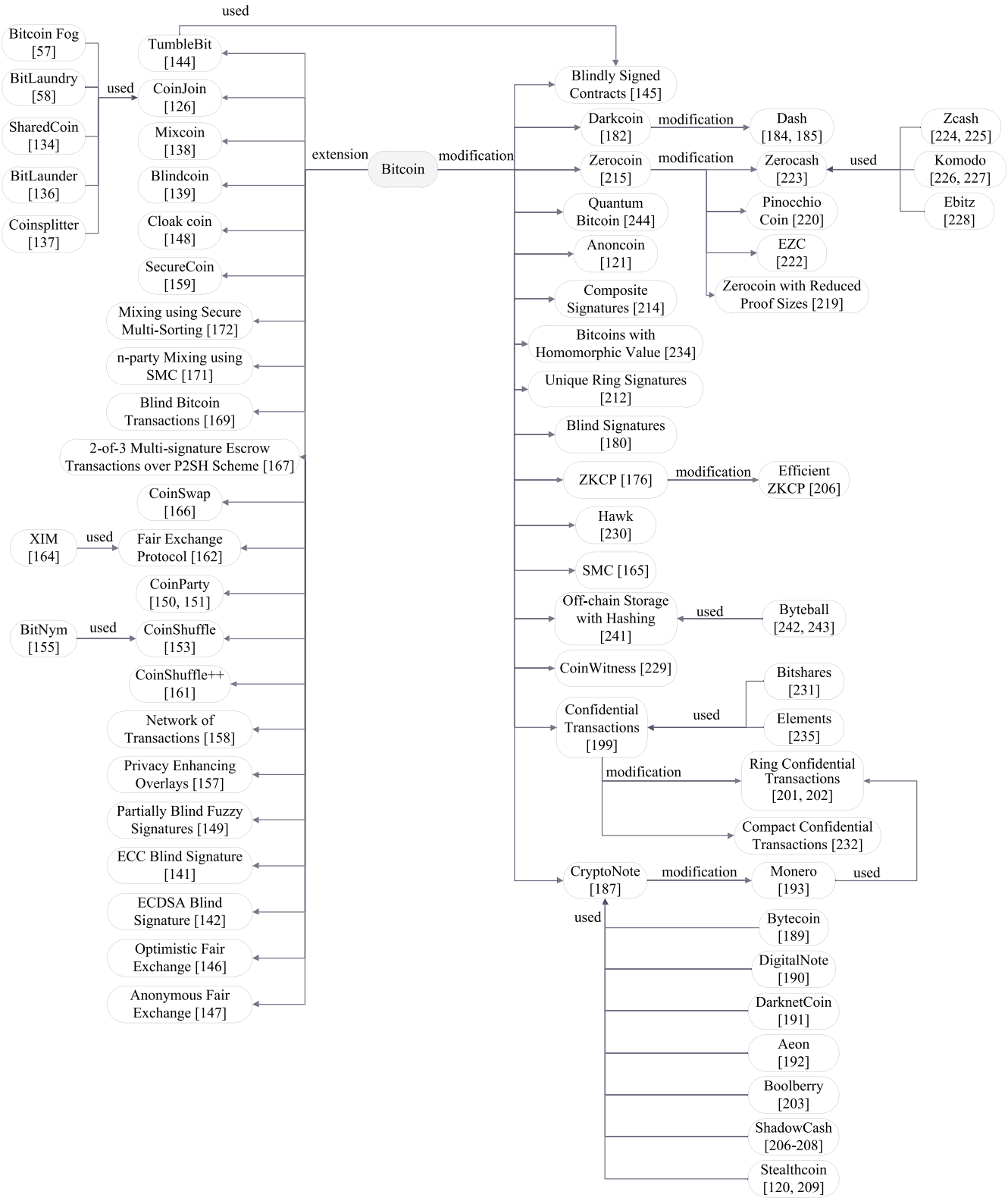

Fig. 12. Relationships between Bitcoin and the anonymity and privacy improving proposals.

section we summarize our observations and lessons learned. Lack of anonymity and privacy in Bitcoin is well-received by everyone. In Bitcoin, flows of cash can be traced, balances of Bitcoin addresses can be calculated. Owners of Bitcoin addresses or Bitcoin addresses of known identities can be discovered by utilizing off-network information. IP addresses related to Bitcoin addresses can be discovered by utilizing the P2P network information like *anomalously relayed transactions*, *first relayer*, *network graph* and *setting address cookie*. Moreover, different Bitcoin addresses owned

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

by the same entity can be linked by analyzing blockchain data and using clustering heuristics. Most used clustering heuristics are (i) assuming that input addresses belong to the same user in a *multi-input transaction* and (ii) one of the output addresses is the *change address* and belong to the user owning the input addresses. Also, *behavior-based clustering techniques*, which use mostly spending habits information, can also be utilized. Nevertheless, users can mix their transactions with transactions of other users by using mixing techniques and outputs of clustering methods would not be consistent with the ground truth for these transactions.

Through examining other characteristic properties of studies on anonymity and privacy analysis in Bitcoin, some findings can also be obtained. Custom Bitcoin clients are built in studies which utilize network in order to map Bitcoin addresses to IP addresses. Degree distribution, clustering coefficient, average shortest path length, and centrality are the network metrics used for the analysis of the P2P network. These metrics are also used for analyzing the transaction and user networks obtained from blockchain data. Metrics to quantify anonymity and privacy provided by the studies are *activity unlinkability* and *profile indistinguishability*. *k-anonymity* metric and taint analysis tool of blockchain.info are also used for measuring anonymity and privacy. There are also some studies proposing privacy-enhancing measures. The simplest and easiest measures are (i) generating a new Bitcoin address for every transaction, (ii) avoiding to reveal any information related to a Bitcoin address in off-network, (iii) avoiding transactions including change, (iv) using modern wallets that do not use Bloom filters. In addition, users must be careful while using mixing servers, since these services may be unreliable or vulnerable to attacks. Network anonymizers should be used in order to avoid mapping IP addresses to Bitcoin addresses.

Findings of studies analyzing anonymity and privacy in Bitcoin brought along the proposals for improving anonymity and privacy. Aims of improvement methods include *hiding amounts*, *breaking the traceability of cash flows* and *hiding IP addresses*. Hiding the amounts provides limited privacy by preventing observers to see the amount and calculate balances, however information of who transferred whom remains public as the links between transactions allow tracing the flow. On the other hand, methods just breaking links between transactions allow observers to retrieve information of separate transactions, although they prevent calculation of balances. Methods breaking links between input and output addresses in a transaction, like CoinJoin, obfuscate the cash flows. However, there is no method to prevent discovering Bitcoin addresses by transacting or discovering identities using off-network information. This is also true for other cryptocurrencies. In addition, behavior-based clustering of Bitcoin addresses cannot be prevented if the amounts are not hidden.

Improvement proposals are either *backwards compatible* or *not backwards compatible*. The main approach in *backwards compatible* proposals is mixing. Mixing can be done either centralized or decentralized and with either *explicit address shuffling* or *hidden address shuffling*. Blind signatures, fair exchange protocols, decryption mixnets, dining cryptographers networks, network of transactions, SMC and zero knowledge proofs are used for *hidden address shuffling* in *the backwards compatible* proposals. The *backwards compatible* proposals are designed for breaking the traceability of cash flows; either *breaking links between input-output addresses in a transaction* or *breaking links between transactions*. *Not backwards compatible* proposals are either new designs improved upon Bitcoin or require a fundamental change in Bitcoin. Main approaches in *not backwards compatible* proposals are *hidden address shuffling*, *ownership shuffling*, *data encrypting* and *data disintegrating*. *Hidden address shuffling* is achieved using blind signatures, ring signatures, and composite signatures. The outcome of *hidden address shuffling* is *breaking links between input-output addresses in a transaction*. *Ownership shuffling* is done using zero knowledge proofs, and the outcome is *breaking links between transactions*, although there are proposals *hiding amounts* too. For *data encrypting*, homomorphic commitments are used and the outcome is *hiding amounts*. *Off-chain storage* is used for *data disintegrating,* and the outcomes are *breaking links between transactions* and *hiding amounts*.

Methods both hiding amounts and breaking traceability of bitcoin flows seem to be the best solution to the problem of anonymity and privacy deficiencies in Bitcoin. These methods are the proposals utilizing zero knowledge proofs, which are *transaction breaker and hiding amounts*, and *off-chain storage*, at least for now. However, the ZK-SNARKs and homomorphic commitments are cryptographically heavy and therefore slow. Proposals utilizing *off-chain storage* seems to be a better option in terms of performance.

Another important point to consider is that the proposals may also have deficits. For instance, although Monero was proposed to fix deficiencies of CryptoNote, it is showed that Monero is also open for attacks [260]; it is traceable, and transaction amounts can be revealed. In time, such deficits of the proposals are fixed, and the protocols are improved according to these findings. Therefore entirely trusting these proposals should be avoided since the proposals are very recent, cryptographic structures used may be immature and need to be scrutinized.

We present guidelines for selecting an approach to design an anonymity and privacy improvement for Bitcoin-like digital cash systems in Table XVII. In this table, we determined four criteria:

- *Bitcoin backwards compatible?:* This criteria shows the preference for Bitcoin backwards compatibility. If it is *yes*, then the approach is suggested according to obtain a system that is backwards compatible with Bitcoin. If it is *no*, then a system that is not backwards compatible with Bitcoin can be obtained by following the recommended approach.
- *Break traceability?:* This criteria shows the preference for preventing traceability of cash flows. If it is *yes*, then the cash flows cannot be traceable when the recommended approach is followed. If it is *no*, then the corresponding approach results in a traceable system.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

TABLE XVII
GUIDELINES FOR DESIGNING AN ANONYMITY/PRIVACY IMPROVEMENT FOR BITCOIN-LIKE DIGITAL CASH SYSTEMS

Bitcoin backwards compatible?	Break traceability?	Hide amounts?	Hide IP addresses?	Approach
Yes	Yes	Yes	Yes	*Not applicable*
Yes	Yes	Yes	No	*Not applicable*
Yes	Yes	No	Yes	Do mixing by explicit/hidden address shuffling AND Use network anonymizers
Yes	Yes	No	No	Do mixing by explicit/hidden address shuffling
Yes	No	Yes	Yes	*Not applicable*
Yes	No	Yes	No	*Not applicable*
Yes	No	No	Yes	Use network anonymizers
Yes	No	No	No	*Not applicable*
No	Yes	Yes	Yes	Shuffle ownership information OR Disintegrate data AND Use network anonymizers
No	Yes	Yes	No	Shuffle ownership information OR Disintegrate data
No	Yes	No	Yes	Do mixing by hidden address shuffling OR Shuffle ownership information OR Disintegrate data AND Use network anonymizers
No	Yes	No	No	Do mixing by hidden address shuffling OR Shuffle ownership information OR Disintegrate data
No	No	Yes	Yes	Encrypt data AND Use network anonymizers
No	No	Yes	No	Encrypt data
No	No	No	Yes	Use network anonymizers
No	No	No	No	*Not applicable*

- *Hide amounts?:* This criteria shows the preference for hiding transaction amounts. If it is *yes*, a system hiding amounts can be obtained by following the corresponding approach. If it is *no,* then the suggested approach gives a system that does not hide the transaction amounts.
- *Hide IP addresses?:* This criteria shows the preference for preventing mapping of IP addresses related to the digital cash addresses. If it is *yes*, the suggested approach gives a system hiding IP addresses. If it is *no,* then the suggested approach gives a system that does not hide the IP addresses.

For each criteria combination, if applicable, we list the possible approaches. For some criteria combinations, there might not be an approach to follow, if those combinations

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

are not feasible. For instance, *hiding amounts* in a *Bitcoin backwards compatible manner* is not applicable (four rows in Table XVII). Moreover, some criteria combinations are meaningless. For example, if breaking traceability, hiding amounts and hiding IP addresses are not desired, then we cannot talk about any privacy improvement (two rows in Table XVII). Remaining rows of the table present applicable guidelines. For instance, for a proposal that is Bitcoin backwards compatible and breaking traceability but not hiding amounts and not hiding IP addresses, the approach can be doing mixing by explicit/hidden address shuffling. In order to achieve a desired level of privacy, the methods used for these approaches until now were presented in Fig. 10. However, researchers can investigate new ways while following these approaches and discover better methods to improve anonymity and privacy.

VI. Future Research Directions

Although Bitcoin market dominance continues, we see that usage of alternative cryptocurrencies, which enhance anonymity and privacy, increases day by day. Dash, Monero, Zcash can be given as example implementations to proposals that are widely recognized. Based on our insights, in a world of digitalization and globalization, we foresee that utilization of digital cash systems will continue to increase and more protocols for improving anonymity and privacy in these systems will emerge with the advances in cryptography and computing. In this regard, we identify four issues that are worthy of further research.

A. Performance

In particular, we conjecture that although many research efforts have been spent on improving performance of anonymity and privacy improving solutions, future research will include investigating more effective methods. Further elaborations are necessary, especially for the proposals that are cryptographically heavy such as zero knowledge proofs and homomorphic commitments to decrease the time required for processing the transactions. Although efficiency improvements are proposed such as a new commitment scheme for Confidential Transactions [261] or a new signature scheme [262] for improving performance of Dash, Monero and Zcash, we believe that there is room for further research.

B. Security

There is a very dynamic research environment on digital cash systems. Number of alternative proposals improving anonymity and privacy in digital cash systems has been increasing rapidly. However, they are very recent and more thorough examination of proposed protocols and cryptographic structures is required. As shown for CryptoNote [196], [198], Monero [260], ZKCP [263] and Zcash [264], these proposals may have vulnerabilities. Therefore, more research efforts should be dedicated to ensuring security of newly proposed cryptographic protocols.

C. Scalability

Another challenge is ensuring scalability while improving anonymity and privacy. Although Bitcoin is the most transparent payment network, researchers continue studies on scalability [265]–[267] and improving anonymity and privacy may bring about more limitations on scalability. Adding additional structures to improve anonymity and privacy brings about larger transaction and blockchain sizes. Although some of the improvement proposals presented scalability analysis of their methods, these analyses may be inadequate when we consider the growing interest in Bitcoin-like digital cash systems, the increasing number of users and transactions. Research continues on scalability, as for ZK-SNARKs [254], [268], Zerocash [269] and Monero [270], and we believe that investigating ways of scalability will continue to take part in further research for new proposals.

D. Anonymity and Trust

Bitcoin addresses are traceable, and transaction amounts are public. These properties enable checking the integrity of the system. Checking integrity of the system becomes harder when measures are taken to improve anonymity and privacy. For instance, as mentioned before, when transaction amounts are hidden, the total number of coins in the system cannot be counted, and if someone breaks the system, he can issue coins without being detected. Similar questions arise when the links between transactions are broken. So when additional mechanisms are added to the system to improve anonymity and privacy, required trust to the system increases. However, users experience difficulty in trusting such systems [271], [272], although they seek privacy. Therefore, we believe balancing anonymity and trust to integrity is worthy of investigation, since users need anonymity and privacy in a digital cash scheme, yet trusting to the integrity of the system, and it is a big challenge to make users trust the system and to provide anonymity and privacy, both at the same time.

VII. Conclusion

In this study, we present a comprehensive survey, which analyzes state of the art anonymity and privacy studies in Bitcoin-like digital cash systems. We classified studies into two main categories; the studies that analyze anonymity and privacy and the studies that propose anonymity and privacy improvements. The first category focuses on revealing information by utilizing blockchain and network analysis, and deanonymization techniques. In this category, we examined and provided a taxonomy for 25 studies, and extracted 9 methods and 5 outcomes from these studies. The aim of the studies is mainly discovering deanonymization techniques and obtaining information that threatens privacy like *discovering Bitcoin addresses*, *discovering identities*, *mapping Bitcoin addresses to IP addresses*, *linking Bitcoin addresses* and *mapping Bitcoin addresses to geo-locations*. Our analysis demonstrates that blockchain analysis covers the majority of the studies in this category and there are few studies on network analysis.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

The most used methods in blockchain analysis are multi-input and change address heuristics, which are used to link Bitcoin addresses that are expected to belong to the same user. Utilizing off-network information is also used extensively. We also analyzed the studies according to some characteristic properties. We observed that actual deanonymization and flow analysis are commonly used in the studies. We also examined the relationship of characteristic properties and methods/outcomes related to the studies analyzing anonymity and privacy.

Examination of the studies that analyze anonymity and privacy in Bitcoin clearly shows that Bitcoin requires anonymity and privacy improvements. As a result, numerous studies exist that include proposals for improving anonymity and privacy in Bitcoin-like digital cash systems. We examined these proposals as the second category. In this category, we provided a taxonomy for 69 studies, extracting 19 methods and 4 outcomes from these studies. *Breaking links between input and output addresses in a transaction*, *breaking links between transactions*, *hiding transaction amounts* and *hiding IP addresses* are the outcomes in this category. Proposals mostly focus on preventing blockchain analysis, although they are suggested to be used with the methods that prevent network analysis. Mixing is the general approach for breaking links of addresses in a transaction or links between transactions, which does not require modification to the Bitcoin protocol. Despite the general assumption of inputs in a multi-input transaction belonging to the same user in the studies of the first category, mixing proposals show that this assumption is not always valid. We also examined the relationship between outcomes of the analyses in Section III and the improvement methods presented in Section IV. We observed that *discovering Bitcoin addresses, discovering identities* and *mapping Bitcoin addresses to geo-locations* cannot be addressed by the methods. However, *mapping Bitcoin addresses to IP addresses* can be prevented by all methods against network analysis, and *linking Bitcoin addresses* can be prevented by all methods against blockchain analysis except homomorphic commitments.

In the proposals that require modification to the protocol, zero knowledge proofs and ring signatures are the mostly used methods for improving anonymity and privacy according to our classification results. Additionally, homomorphic commitments are used for hiding transaction amounts as an alternative to zero knowledge protocols. Our further discussions include relationships between Bitcoin and the anonymity and privacy improving proposals, and comparison of the methods from the performance perspective.

In Sections V and VI, we summarized our observations and lessons learned and then provided future research directions. We believe that there is room for more research on performance, security, scalability, anonymity and trust. The necessity of modification and poor performance due to increasing cryptographic elements are the issues that make the proposals difficult to be adopted, and there are still opportunities for further enhancements. The number of alternative protocols that utilize off-chain storage, which give promise of good performance and scalability, also increases, including proposals that are not applicable yet, like the ones utilizing quantum computers. It should be noted that the methods that improve anonymity and privacy decrease the possibility of checking the integrity of the system as a whole, resulting the increase of required trust to the system.

REFERENCES

[1] S. Nakamoto. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[2] E. Harper, *'I Didn't Buy That': Friendly Fraud Costs Retailers $11.8 Billion a Year*, AOL, New York, NY, USA, Mar. 2014, accessed: Nov. 1, 2017. [Online]. Available: https://www.aol.com/article/finance/2014/03/20/friendly-fraud-costs-retailers-billions/20853307

[3] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology: Proceedings of Crypto 82*. Boston, MA, USA: Springer, pp. 199–203, 1983.

[4] *Satoshi Nakamoto, Wikipedia*. Accessed: Jul. 10, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Satoshi_Nakamoto

[5] *Bitcoin Block Explorer*. Accessed: Jun. 13, 2017. [Online]. Available: https://blockexplorer.com/blocks-date/[year-month-day]

[6] A. Chen, *We Need To Know Who Satoshi Nakamoto Is*, The New Yorker, New York, NY, USA, May 2016, accessed: Jul. 10, 2016. [Online]. Available: http://www.newyorker.com/business/currency/we-need-to-know-who-satoshi-nakamoto-is

[7] *Has Craig Wright Proved He's Bitcoin's Satoshi Nakamoto?* BBC News, London, U.K., 2016, accessed: Jul. 10, 2016. [Online]. Available: http://www.bbc.com/news/technology-36191165

[8] *CryptoCurrency Market Capitalizations*. Accessed: Dec. 15, 2017. [Online]. Available: https://coinmarketcap.com

[9] *Bitcoin Charts / Bitcoin Network, Bitcoin Charts*. Accessed: Dec. 15, 2017. [Online]. Available: http://bitcoincharts.com/bitcoin/

[10] *Total Number of Transactions*, Blockchain, Luxembourg City, Luxembourg, accessed: Jun. 13, 2017. [Online]. Available: https://blockchain.info/charts/n-transactions-total

[11] T. Schaffner. (2014). *Bitcoin Anonymity and Security*. Accessed: Jan. 9, 2017. [Online]. Available: http://www.cs.tufts.edu/comp/116/archive/fall2014/tschaffner.pdf

[12] Q. ShenTu and J. Yu, "Research on anonymization and deanonymization in the Bitcoin system," *arXiv preprint arXiv:1510.07782*, Oct. 2015. [Online]. Available: https://arxiv.org/abs/1510.07782

[13] J. Herrera-Joancomartí, "Research and challenges on Bitcoin anonymity," in *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance*. Cham, Switzerland: Springer, 2015, pp. 3–16.

[14] J. Bonneau *et al.*, "SoK: Research perspectives and challenges for Bitcoin and cryptocurrencies," in *Proc. IEEE Symp. Security Privacy*, San Jose, CA, USA, 2015, pp. 104–121.

[15] A. Narayanan *et al.*, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton, NJ, USA: Princeton Univ. Press, 2016.

[16] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2084–2123, 3rd Quart., 2016.

[17] F. K. Maurer, *A Survey on Approaches to Anonymity in Bitcoin and Other Cryptocurrencies* (Lecture Notes in Informatics), vol. 259. Bonn, Germany: Gesellschaft für Informatik, 2016, pp. 2145–2150.

[18] V. Buterin, *Ethereum White Paper: A Next Generation Smart Contract & Decentralized Application Platform*. Accessed: Nov. 19, 2017. [Online]. Available: https://www.weusecoins.com/assets/pdf/library/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin.pdf

[19] G. Wood. *Ethereum: A Secure Decentralised Generalised Transaction Ledger*. Accessed: Nov, 19, 2016. [Online]. Available: http://paper.gavwood.com

[20] S. Tikhomirov, "Ethereum: State of knowledge and research perspectives," in *Proc. 10th Int. Symp. Found. Pract. Security*, 2017, pp. 206–221.

[21] Ethereum Blog. (Jan. 2016). *Privacy on the Blockchain*. Accessed: Jan. 13, 2017. [Online]. Available: https://blog.ethereum.org/2016/01/15/privacy-on-the-blockchain

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

[22] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Succinct non-interactive zero knowledge for a von Neumann architecture," IACR Cryptol. ePrint Archive, Rep. 2013/879, Dec. 2013. [Online]. Available: https://eprint.iacr.org/2013/879

[23] E. Ben-Sasson, A. Chiesa, D. Genkin, E. Tromer, and M. Virza, "SNARKs for C: Verifying program executions succinctly and in zero knowledge," in *Advances in Cryptology—CRYPTO 2013* (LNCS 8043). Heidelberg, Germany: Springer, 2013, pp. 90–108.

[24] Ethereum Blog. *An Update on Integrating Zcash on Ethereum (ZoE)*. Accessed: Feb. 13, 2017. [Online]. Available: https://blog.ethereum.org/2017/01/19/update-integrating-zcash-ethereum

[25] Z. Wilcox, *Ethereum Adoption of zk-SNARK Technology*, Zcash Blog, Sep. 2017, accessed: Nov. 20, 2017. [Online]. Available: https://z.cash/blog/ethereum-snarks.html

[26] D. Bradbury, "Anonymity and privacy: A guide for the perplexed," *Netw. Security*, vol. 2014, no. 10, pp. 10–14, Oct. 2014.

[27] D. Eckhoff and I. Wagner, "Privacy in the smart city—Applications, technologies, challenges, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 489–516, 1st Quart., 2018.

[28] Z. Xiao, and Y. Xiao, "Security and privacy in cloud computing," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 843–859, 2nd Quart., 2013.

[29] M. A. Ferrag, L. Maglaras, and A. Ahmim, "Privacy-preserving schemes for ad hoc social networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 3015–3045, 4th Quart., 2017.

[30] D. Davenport, "Anonymity on the Internet: Why the price may be too high," *Commun. ACM*, vol. 45, no. 4, pp. 33–35, Apr. 2002.

[31] D. Kelly, R. Raines, R. Baldwin, M. Grimaila, and B. Mullins, "Exploring extant and emerging issues in anonymous networks: A taxonomy and survey of protocols and metrics," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 2, pp. 579–606, 2nd Quart., 2012.

[32] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Commun. ACM*, vol. 24, no. 2, pp. 84–90, Feb. 1981.

[33] D. Chaum *et al.*, "cMix: Anonymization by high-performance scalable mixing," IACR Cryptol. ePrint Archive, Rep. 2016/008, Jan. 2016. [Online]. Available: https://eprint.iacr.org/2016/008

[34] M. G. Reed, P. F. Syverson, and D. M. Goldschlag, "Anonymous connections and onion routing," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 4, pp. 482–494, May 1998.

[35] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," in *Proc. 13th Conf. USENIX Security Symp. (SSYM)*, vol. 13. San Diego, CA, USA, 2004, pp. 21–37.

[36] E. Erdin, C. Zachor, and M. H. Gunes, "How to find hidden users: A survey of attacks on anonymity networks," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2296–2316, 4th Quart., 2015.

[37] (Mar. 2017). *A Taxi Driver Registered in 'Bitaksi' Application Plans to Murder a Passenger After Very Deserved Bad Review, Reddit: The Front Page of the Internet*. Accessed: Jan. 23, 2018. [Online]. Available: https://redd.it/61zczy

[38] R. C. Merkle, "A digital signature based on a conventional Encryption function," in *Advances in Cryptology—CRYPTO '87* (Lecture Notes in Computer Science), vol. 293. Heidelberg, Germany: Springer, 1988, pp. 369–378.

[39] Bitcoin Developer Guide. *Bitcoin—Open source P2P Money*. Accessed: Jun. 1, 2016. [Online]. Available: https://bitcoin.org/en/developer-guide

[40] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ECDSA)," *Int. J. Inf. Security*, vol. 1, no. 1, pp. 36–63, Aug. 2001.

[41] A. Back. (Aug. 2002). *Hashcash—A Denial of Service Counter-Measure*. Accessed: Jul. 6, 2016. [Online]. Available: http://www.hashcash.org/hashcash.pdf

[42] S. Nakamoto. (Nov. 2008). *Re: Bitcoin P2P e-cash Paper, The Mail Archive*. Accessed: Oct. 26, 2017. [Online]. Available: https://www.mail-archive.com/cryptography@metzdowd.com/msg09997.html

[43] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, pp. 382–401, 1982.

[44] Bitcoin. *Some Things You Need to Know*. Accessed: Jun. 13, 2016. [Online]. Available: https://bitcoin.org/en/you-need-to-know

[45] Bitcoin. *Protect Your Privacy*. Accessed: Jun. 13, 2016. [Online]. Available: https://bitcoin.org/en/protect-your-privacy

[46] *OBPP Bitcoin Wallet Privacy Rating Report 2nd Edition*. Accessed: Nov. 14. 2016. [Online]. Available: https://raw.githubusercontent.com/OpenBitcoinPrivacyProject/wallet-ratings/master/report-02/OBPP%20Bitcoin%20Wallet%20Privacy%20Rating%20Report%202nd%20Edition%20-%20March%202016.pdf

[47] G. Zyskind, O. Nathan, and A. Pentland, "Enigma: Decentralized computation platform with guaranteed privacy," *arXiv preprint arXiv:1506.03471*, Jun. 2015. [Online]. Available: https://arxiv.org/abs/1506.03471

[48] L. Xu *et al.*, "Enabling the sharing economy: Privacy respecting contract based on public blockchain," in *Proc. ACM Workshop Blockchain Cryptocurrencies Contracts*, 2017, pp. 15–21.

[49] J. P. Morgan. *Quorum*. Accessed: Dec. 13, 2017. [Online]. Available: https://www.jpmorgan.com/global/Quorum

[50] GitHub. *Quorum Whitepaper*. Accessed: Dec. 13, 2017. [Online]. Available: https://github.com/jpmorganchase/quorum-docs/blob/master/Quorum%20Whitepaper%20v0.1.pdf

[51] *Hyperledger Fabric FAQ*. Accessed: Dec. 13, 2017. [Online]. Available: http://hyperledger-fabric.readthedocs.io/en/release/Fabric-FAQ.html?highlight=privacy

[52] F. Reid and M. Harrigan, "An analysis of anonymity in the Bitcoin system," in *Security and Privacy in Social Networks*. New York, NY, USA: Springer, 2012, pp. 197–223.

[53] S. Meiklejohn *et al.*, "A fistful of Bitcoins: Characterizing payments among men with no names," in *Proc. ACM Conf. Internet Meas. Conf. (IMC)*, Barcelona, Spain, 2013, pp. 127–140.

[54] Mt. Gox. *Bitcoin Wiki*. Accessed: Oct. 12, 2016. [Online]. Available: https://en.bitcoin.it/wiki/Mt._Gox

[55] Wikipedia. *Silk Road (Marketplace)*. Accessed: Oct. 12, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Silk_Road_(marketplace)

[56] M. Möser, R. Böhme, and D. Breuker, "An inquiry into money laundering tools in the Bitcoin ecosystem," in *Proc. eCrime Researchers Summit (eCRS)*, San Francisco, CA, USA, 2013, pp. 1–14.

[57] Akemashite Omedetou. (Oct. 2011). *[ANNOUNCE] Bitcoin Fog: Secure Bitcoin Anonymization, Bitcoin Forum*. Accessed: Oct. 14, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=50037

[58] BitLaundry. *Bitcoin Wiki*. Accessed: Oct. 14, 2016. [Online]. Available: https://en.bitcoin.it/wiki/BitLaundry

[59] *Blockchain.info Bitcoin Wallet—Features, Bitcoin Block Explorer—Blockchain*. Accessed: Oct. 14, 2016. [Online]. Available: https://blockchain.info/wallet/features

[60] *Bitcoin Block Explorer—Blockchain*. Accessed: Oct. 14, 2016. [Online]. Available: https://blockchain.info

[61] F. Reid and M. Harrigan, "An analysis of anonymity in the Bitcoin system," in *Proc. IEEE Int. Conf. Privacy Security Risk Trust IEEE Int. Conf. Soc. Comput.*, Boston, MA, USA, 2011, pp. 1318–1326.

[62] D. Ron and A. Shamir, "Quantitative analysis of the full Bitcoin transaction graph," in *Financial Cryptography and Data Security* (LNCS 7859). Heidelberg, Germany: Springer, 2013, pp. 6–24.

[63] *WikiLeaks*. Accessed: Oct. 20, 2016. [Online]. Available: https://wikileaks.org

[64] M. S. Ortega, "The Bitcoin transaction graph anonimity," M. S. thesis, Security Inf. Commun. Technol., Universitat Autònoma de Barcelona, Barcelona, Spain, 2013, accessed: Sep. 14, 2016. [Online]. Available: http://openaccess.uoc.edu/webapps/o2/bitstream/10609/23562/9/msantamariaoTFM0613memoria.pdf

[65] *Bitcoin Forum*. accessed: Oct. 10, 2016. [Online]. Available: https://bitcointalk.org

[66] *Bitcoin Address Tags—Blockchain.info. Bitcoin Block Explorer—Blockchain*. Accessed: Oct. 12, 2016. [Online]. Available: http://blockchain.info/tags

[67] M. Fleder, M. S. Kester, and S. Pillai, "Bitcoin transaction graph analysis," *arXiv preprint arXiv:1502.01657*, Feb. 2015. [Online]. Available: https://arxiv.org/abs/1502.01657

[68] M. Spagnuolo, F. Maggi, and S. Zanero, "BitIodine: Extracting intelligence from the Bitcoin network," in *Financial Cryptography and Data Security* (LNCS 8437). Heidelberg, Germany: Springer, 2014, pp. 457–468.

[69] *#bitcoin-otc*. Accessed: Oct. 10, 2016. [Online]. Available: https://bitcoin-otc.com

[70] *Physical Bitcoins by Casascius*. Accessed: Oct. 10, 2016. [Online]. Available: https://www.casascius.com

[71] *Bitcoin Stock Exchange BitFunder Announces Closure, Bitcoin News, Blockchain News, Prices, Charts & Analysis—CoinDesk*. Accessed: Oct. 11, 2016. [Online]. Available: http://www.coindesk.com/bitcoin-stock-exchange-bitfunder-announces-closure

[72] Wikipedia. *CryptoLocker*. Accessed: Oct. 12, 2016. [Online]. Available: https://en.wikipedia.org/wiki/CryptoLocker

[73] *GitHub—mikispag/bitiodine*. Accessed: Nov. 21, 2016. [Online]. Available: https://github.com/mikispag/bitiodine

[74] *GitHub—mikispag/rusty-blockparser*. Accessed: Nov. 21, 2016. [Online]. Available: https://github.com/mikispag/rusty-blockparser

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

[75] A. Baumann, B. Fabian, and M. Lischke, "Exploring the Bitcoin network," in *Proc. 10th Int. Conf. Web Inf. Syst. Technol.*, vol. 1. Barcelona, Spain, 2014, pp. 369–374.

[76] *Global Bitcoin Nodes Distribution—Bitnodes*. Accessed: Oct. 13, 2016. [Online]. Available: https://getaddr.bitnodes.io

[77] A. Biryukov, D. Khovratovich, and I. Pustogarov, "Deanonymisation of clients in Bitcoin P2P network," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, Scottsdale, AZ, USA, 2014, pp. 15–29.

[78] M. Lischke and B. Fabian, "Analyzing the Bitcoin network: The first four years," *Future Internet*, vol. 8, no. 1, p. 7, Jul. 2016.

[79] *IP Address Details*. Accessed: Jan. 7, 2017. [Online]. Available: http://ipinfo.io

[80] *Tor Network Status*. Accessed: Feb. 8, 2017. [Online]. Available: https://torstatus.blutmagie.de

[81] *TOR List*. Accessed: Feb. 8, 2017. [Online]. Available: https://dan.me.uk/torlist

[82] *Proxy List*. Accessed: Feb. 8, 2017. [Online]. Available: https://vpngeeks.com/proxylist

[83] P. Koshy, D. Koshy, and P. Mcdaniel, "An analysis of anonymity in Bitcoin using P2P network traffic," in *Financial Cryptography and Data Security* (LNCS 8437). Heidelberg, Germany: Springer, 2014, pp. 469–485.

[84] D. Kaminsky, *Black Ops of TCP/IP 2011*, Dan Kaminsky's Blog, Aug. 2011, accessed: Nov. 17, 2016. [Online]. Available: https://dankaminsky.com/2011/08/05/bo2k11/

[85] BlitCoin. *Unmasks One or Both Ends of a Bitcoin Transaction? Bitcoin Forum*. Accessed: Oct. 28, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=34383.0

[86] G. Fanti and P. Viswanath, "Anonymity properties of the Bitcoin P2P network," *arXiv preprint arXiv: 1703.08761*, Mar. 2017. [Online]. Available: https://arxiv.org/abs/1703.08761

[87] T. Neudecker and H. Hartenstein, "Could network information facilitate address clustering in Bitcoin?" in *Proc. Int. Conf. Financ. Cryptograp. Data Security*, 2017, pp. 155–169.

[88] S. Feld, M. Schönfeld, and M. Werner, "Analyzing the deployment of Bitcoin's P2P network under an AS-level perspective," *Procedia Comput. Science*, vol. 32, pp. 1121–1126, Jun. 2014.

[89] P. Golle and A. Juels, "Dining cryptographers revisited," in *Proc. Adv. Cryptol. EUROCRYPT*, 2004, pp. 456–473.

[90] A. Biryukov and I. Pustogarov, "Bitcoin over Tor isn't a good idea," in *Proc. IEEE Symp. Security Privacy*, San Jose, CA, USA, 2015, pp. 122–134.

[91] *Bitcoin Core*. Accessed: Oct. 13, 2016 [Online]. Available: https://bitcoin.org/en/bitcoin-core

[92] D. Ron and A. Shamir, "How did dread pirate Roberts acquire and protect his Bitcoin wealth?" in *Financial Cryptography and Data Security* (LNCS 8438). Heidelberg, Germany: Springer, 2014, pp. 3–15.

[93] E. Androulaki, G. O. Karame, M. Roeschlin, T. Scherer, and S. Capkun, "Evaluating user privacy in Bitcoin," in *Financial Cryptography and Data Security* (LNCS 7859). Heidelberg, Germany: Springer, 2013, pp. 34–51.

[94] M. Ober, S. Katzenbeisser, and K. Hamacher, "Structure and anonymity of the Bitcoin transaction graph," *Future Internet*, vol. 5, no. 2, pp. 237–250, May 2013.

[95] J. Dupont and A. C. Squicciarini, "Toward de-anonymizing Bitcoin by mapping users location," in *Proc. 5th ACM Conf. Data Appl. Security Privacy*, San Antonio, TX, USA, 2015, pp. 139–141.

[96] C. Zhao and Y. Guan, "A graph-based investigation of Bitcoin transactions," in *IFIP Advances in Information and Communication Technology Advances in Digital Forensics XI*, vol. 462. Cham, Switzerland: Springer, 2015, pp. 79–95.

[97] C. Zhao, "Graph-based forensic investigation of Bitcoin transactions," M.S. thesis, Dept. Comput. Eng., Iowa State Univ., Ames, IA, USA, 2014, accessed: Jan. 7, 2017. [Online]. Available: http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=5253&context=etd

[98] J. D. Nick, "Data-driven de-anonymization in Bitcoin," M.S. thesis, Distrib. Comput. Group Comput. Eng. Netw. Lab., ETH Zürich, Zürich, Switzerland, 2015. accessed: Jan. 16, 2017. [Online]. Available: http://e-collection.library.ethz.ch/eserv/eth:48205/eth-48205-01.pdf

[99] D. Ferrin. (Mar. 2015). *A Preliminary Field Guide for Bitcoin Transaction Patterns*. Accessed: Jan. 13, 2017. [Online]. Available: https://www.smithandcrown.com/open-research/a-preliminary-field-guide-for-bitcoin-transaction-patterns

[100] Y. Yanovich, P. Mischenko, and A. Mischenko. (Aug. 2016). *Shared Send Untangling in Bitcoin Whitepaper*. Accessed: Jan. 3, 2017. [Online]. Available: https://bravenewcoin.com/assets/Whitepapers/bitfury-whitepaper-shared-send-untangling-in-bitcoin-8-24-2016.pdf

[101] *BCHAIN |Bitcoin Number of Unique Bitcoin Addresses Used*. Accessed: Jun. 13, 2017. [Online]. Available: https://www.quandl.com/data/BCHAIN/NADDU-Bitcoin-Number-of-Unique-Bitcoin-Addresses-Used

[102] *GitHub—Ivan-Brugere/Bitcoin-Transaction-Network-Extraction*. Accessed: Nov. 17, 2016. [Online]. Available: https://github.com/ivan-brugere/Bitcoin-Transaction-Network-Extraction/blob/master/README.pdf

[103] *GitHub—Harrigan/Bitcointools: Python-Based Tools for the Bitcoin Cryptocurrency System*. Accessed: Nov. 21, 2016. [Online]. Available: github.com/harrigan/bitcointools

[104] *GitHub—Gavinandresen/Bitcointools: Python-Based Tools for the Bitcoin Cryptocurrency System*. Accessed: Nov. 21, 2016. [Online]. Available: github.com/gavinandresen/bitcointools

[105] *Bloom Filter—Bitcoin Glossary, Bitcoin—Open Source P2P Money*. Accessed: Dec. 30, 2016. [Online]. Available: https://bitcoin.org/en/glossary/bloom-filter

[106] L. Sweeney, "K-anonymity: A model for protecting privacy," *Int. J. Uncertainty Fuzziness Knowl. Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[107] *Bitcoin Taint Analysis, Bitcoin Block Explorer—Blockchain*. Accessed: Nov. 8, 2016. [Online]. Available: http://blockchain.info/de/taint/_ADDRESS_

[108] *TorBan—Stats on Tor Exit Nodes Used to Connect to the Bitcoin Network*. Accessed: Jan. 24, 2017. [Online]. Available: http://www.openbitcoinprivacyproject.org/torban/

[109] *Fallback Nodes, Bitcoin Wiki*. Accessed: Nov. 23, 2016. [Online]. Available: https://en.bitcoin.it/wiki/Fallback Nodes

[110] *Slush Pool*. Accessed: Oct. 14, 2016. [Online]. Available: https://slushpool.com/home

[111] *LulzSec, Wikipedia*. Accessed: Oct. 14, 2016. [Online]. Available: https://en.wikipedia.org/wiki/LulzSec

[112] *DeepBit, Bitcoin Wiki*. Accessed: Oct. 13, 2016. [Online]. Available: https://en.bitcoin.it/wiki/DeepBit

[113] *Satoshi Dice, Bitcoin Wiki*. Accessed: Oct. 15, 2016. [Online]. Available: https://en.bitcoin.it/wiki/Satoshi_Dice

[114] (Jun. 2011). *Allinvain: I Just Got Hacked—Any Help is Welcome! (25,000 BTC Stolen), Bitcoin Forum*. Accessed: Nov. 8, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=16457.0

[115] *List of Major Bitcoin Heists, Thefts, Hacks, Scams, and Losses [Old], Bitcoin Forum*. Accessed: Nov. 16, 2016. [Online]. Available: https://bitcointalk.org/index.php? topic=83794.0

[116] D. Kondor, M. Pósfai, I. Csabai, and G. Vattay, "Do the rich get richer? An empirical analysis of the BitCoin transaction network," *arXiv preprint arXiv: 1308.3892*, Mar. 2014. [Online]. Available: https://arxiv.org/abs/1308.3892

[117] R. Dorfman, "A formula for the gini coefficient," *Rev. Econ. Stat.*, vol. 61, no. 1, pp. 146–149, 1979.

[118] J. A. D. Donet *et al.*, "The bitcoin P2P network," in *Financial Cryptography and Data Security* (LNCS 8438). Heidelberg, Germany: Springer, 2014, pp. 87–102.

[119] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of peer-to-peer overlay network schemes," *IEEE Commun. Surveys Tuts.*, vol. 7, no. 2, pp. 72–93, 2nd Quart., 2005.

[120] *StealthCoin*. Accessed: Jan. 23, 2017. [Online]. Available: https://www.stealth-coin.com/

[121] (May 2016). *Cryptoslave: [ANN][ANC] Anoncoin (anon-coin.net) |Privacy-Centric Currency |I2P Darknet, Bitcoin Forum*. Accessed: Jan. 3, 2017. [Online]. Available: https://bitcointalk.org/index.php?topic=1481693.0

[122] F. Astolfi, J. Kroese, and J. V. Oorschot. *I2P—The Invisible Internet Project*. Accessed: Feb. 17, 2017. [Online]. Available: http://mediatechnology.leiden.edu/images/uploads/docs/wt2015_i2p.pdf

[123] (Mar. 2013). *GIV: Bitcoin Client With I2P Patch, Bitcoin Forum*. Accessed: Jan. 22, 2017. [Online]. Available: https://bitcointalk.org/index.php?topic=151181

[124] *Tutorial: How To Access Exchanged.i2p, Deep Dot Web*. Accessed: Jan. 24, 2017. [Online]. Available: https://www.deepdotweb.com/2016/05/17/tutorial-access-exchanged-i2p/

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

[125] Q. ShenTu and J. Yu, "Transaction remote release (TRR): A new anonymization technology for bitcoin," *arXiv preprint arXiv: 1509.06160*, Sep. 2015. [Online]. Available: https://arxiv.org/abs/1509.06160

[126] (Aug. 2013). *Gmaxwell: CoinJoin: Bitcoin Privacy for the Real World, Bitcoin Forum*. Accessed: Jan. 12, 2017. [Online]. Available: https://bitcointalk.org/index.php?topic=279249.0

[127] Gmaxwell. (Aug. 2013). *CoinJoin: Bitcoin Privacy for the Real World (Someday!) Bitcoin Forum*. Accessed: Dec. 24, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=279249.msg2984051#msg2984051

[128] *GitHub—Michaelgpearce/Coinmux: Decentralized, Trustless, Anonymous and Open Bitcoin Mixer*. Accessed: Jan. 12, 2017. [Online]. Available: https://github.com/michaelgpearce/coinmux

[129] *GitHub—Chris-Belcher/Coinjumble: GUI for Doing Bitcoin Coinjoin in An Asynchronous Manner*. Accessed: Jan. 12, 2017. [Online]. Available: https://github.com/chris-belcher/coinjumble

[130] *The SX Tutorial—SX 1 Documentation*. Accessed: Jan. 27, 2017. [Online]. Available: http://sx.dyne.org

[131] *Dark Wallet*. Accessed: Jan. 10, 2017. [Online]. Available: https://www.darkwallet.is

[132] *Dark Wallet Overview*. Accessed: Jan. 10, 2017. [Online]. Available: https://wiki.unsystem.net/en/index.php/DarkWallet/Overview

[133] *Samourai Bitcoin Wallet—Features*. Accessed: Dec. 27, 2016. [Online]. Available: http://samouraiwallet.com/features.html

[134] *Shared Coin, Bitcoin Wiki*. Accessed: Sep. 12, 2016. [Online]. Available: https://en.bitcoin.it/wiki/Shared_coin

[135] *CoinJoin Sudoku |Weaknesses in SharedCoin, and CoinJoin Research*. Accessed: Jan. 19, 2017. [Online]. Available: http://www.coinjoinsudoku.com

[136] *BitLaunder*. Accessed: Nov. 3, 2016. [Online]. Available: https://bitlaunder.com

[137] *CoinSplitter*. Accessed: Nov. 3, 2016. [Online]. Available: https://coinsplitter.org/

[138] J. Bonneau *et al.*, "Mixcoin: Anonymity for bitcoin with accountable mixes," in *Financial Cryptography and Data Security* (LNCS 8437). Heidelberg, Germany: Springer, 2014, pp. 486–504.

[139] L. Valenta and B. Rowan, "Blindcoin: Blinded, accountable mixes for bitcoin," in *Financial Cryptography and Data Security* (LNCS 8976). Heidelberg, Germany: Springer, 2015, pp. 112–126.

[140] G. Fuchsbauer, "Automorphic signatures in bilinear groups and an application to round-optimal blind signatures," IACR Cryptol. ePrint Archive, Rep. 2009/320, Jun. 2009. [Online]. Available: https://eprint.iacr.org/2009/320

[141] Q. ShenTu and J. Yu, "A blind-mixing scheme for Bitcoin based on an elliptic curve cryptography blind digital signature algorithm," *arXiv preprint arXiv: 1510.05833*, Oct. 2015. [Online]. Available: https://arxiv.org/abs/1510.05833

[142] O. Andreev. (Feb. 2014). *Blind Signatures for Bitcoin Transactions*. Accessed: Jan. 15, 2017. [Online]. Available: http://oleganza.com/blind-ecdsa-draft-v2.pdf

[143] M. Svhunter, "Fair exchange," in *Encyclopedia of Cryptography and Security*. Boston, MA, USA: Springer, 2011, pp. 447–448.

[144] E. Heilman, L. Alshenibr, F. Baldimtsi, A. Scafuro, and S. Goldberg, "TumbleBit: An untrusted bitcoin-compatible anonymous payment hub," IACR Cryptol. ePrint Archive, Rep. 2016/575, Jun. 2016. [Online]. Available: https://eprint.iacr.org/2016/575

[145] E. Heilman, F. Baldimtsi, and S. Goldberg, "Blindly signed contracts anonymous on-blockchain and off-blockchain bitcoin transactions," IACR Cryptol. ePrint Archive, Rep. 2016/056, Jan. 2016. [Online]. Available: https://eprint.iacr.org/2016/056

[146] D. Jayasinghe, K. Markantonakis, and K. Mayes, "Optimistic fair-exchange with anonymity for bitcoin users," in *Proc. IEEE 11th Int. Conf. e-Bus. Eng. (ICEBE)*, Guangzhou, China, 2014, pp. 44–51.

[147] D. Jayasinghe and K. Jayasinghe, *Digital Cash and Anonymous Fair-Exchange Payment Protocols*, Computer Weekly, London, U.K., Sep. 2016, accessed: Feb. 1, 2017. [Online]. Available: https://www.royalholloway.ac.uk/isg/documents/pdf/technicalreports/2016/computer-weekly-articles/danushkajayasinghecw.pdf

[148] (Apr. 2015). *A Trustless, Anonymous Transaction System for CloakCoin*. Accessed: Dec. 12, 2016. [Online]. Available: https://bravenewcoin.com/assets/Whitepapers/CloakCoin-posa3wp.pdf

[149] Q. Wu *et al.*, "Secure joint Bitcoin trading with partially blind fuzzy signatures," *Soft Comput.*, vol. 21, no. 11, pp. 3123–3134, 2015.

[150] J. H. Ziegeldorf, F. Grossmann, M. Henze, N. Inden, and K. Wehrle, "CoinParty: Secure multi-party mixing of bitcoins," in *Proc. 5th ACM Conf. Data Appl. Security Privacy (CODASPY)*, San Antonio, TX, USA, 2015, pp. 75–86.

[151] J. H. Ziegeldorf, R. Matzutt, M. Henze, F. Grossmann, and K. Wehrle, "Secure and anonymous decentralized Bitcoin mixing," *Future Gener. Comput. Syst.*, vol. 80, pp. 448–466, Mar. 2018.

[152] G. Bleumer, "Threshold signature," in *Encyclopedia of Cryptography and Security*. Boston, MA, USA: Springer, 2011, pp. 1294–1296.

[153] T. Ruffing, P. Moreno-Sanchez, and A. Kate, "CoinShuffle: Practical decentralized coin mixing for Bitcoin," in *Computer Security—ESORICS 2014* (LNCS 8713). Cham, Switzerland: Springer, 2014, pp. 345–364.

[154] H. Corrigan-Gibbs and B. Ford, "Dissent: Accountable anonymous group messaging," in *Proc. 17th ACM Conf. Comput. Commun. Security (CCS)*, Chicago, IL, USA, 2010, pp. 340–350.

[155] M. Florian, J. Walter, and I. Baumgart, "Sybil-resistant pseudonymization and pseudonym change without trusted third parties," in *Proc. 14th ACM Workshop Privacy Electron. Soc. (WPES)*, Denver, CO, USA, 2015, pp. 65–74.

[156] *CoinShuffle: Practical Decentralized Bitcoin Mixing*, CrypSys Cryptograph. Syst. Res. Group, Saarbrücken, Germany, accessed: Dec. 20, 2016. [Online]. Available: http://crypsys.mmci.uni-saarland.de/projects/CoinShuffle

[157] S. Meiklejohn and C. Orlandi, "Privacy-enhancing overlays in bitcoin," in *Financial Cryptography and Data Security* (LNCS 8976). Heidelberg, Germany: Springer, 2015, pp. 127–141.

[158] O. Coutu, "Privacy in bitcoin through decentralized mixers," M. S. thesis, Département d'informatique et de Recherche Opérationnelle, Université de Montréal, Montreal, QC, Canada, Apr. 2014, accessed: Dec. 28, 2016. [Online]. Available: https://papyrus.bib.umontreal.ca/xmlui/bitstream/handle/1866/11498/Coutu_Olivier_2014_memoire.pdf

[159] M. H. Ibrahim, "SecureCoin: A robust secure and efficient protocol for anonymous bitcoin ecosystem," *Int. J. Netw. Security*, vol. 19, no. 2, pp. 295–312, Mar. 2017.

[160] D. Chaum, "The dining cryptographers problem: Unconditional sender and recipient untraceability," *J. Cryptol.*, vol. 1, no. 1, pp. 65–75, Jan. 1988.

[161] T. Ruffing, P. Moreno-Sanchez, and A. Kate, "P2P mixing and unlinkable bitcoin transactions," IACR Cryptol. ePrint Archive, Rep. 2016/824, Aug. 2016. [Online]. Available: https://eprint.iacr.org/2016/824

[162] S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better—How to make bitcoin a better currency," in *Financial Cryptography and Data Security* (LNCS 7397). Heidelberg, Germany: Springer, 2012, pp. 399–414.

[163] C. Crépeau, "Cut-and-choose protocol," in *Encyclopedia of Cryptography and Security*. Boston, MA, USA: Springer, 2011, pp. 290–291.

[164] G. Bissias, A. P. Ozisik, B. N. Levine, and M. Liberatore, "Sybil-resistant mixing for bitcoin," in *Proc. 13th Workshop Privacy Electron. Soc. (WPES)*, Scottsdale, AZ, USA, 2014, pp. 149–158.

[165] M. Andrychowicz, S. Dziembowski, D. Malinowski, and L. Mazurek, "Secure multiparty computations on bitcoin," in *Proc. IEEE Symp. Security Privacy (SP)*, San Jose, CA, USA, 2014, pp. 443–458.

[166] *CoinSwap: Transaction Graph Disjoint Trustless Trading, Bitcoin Forum*. Accessed: Nov. 17, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=321228

[167] D. A. Wijaya, J. K. Liu, R. Steinfeld, S. F. Sun, and X. Huang, "Anonymizing bitcoin transaction," in *Information Security Practice and Experience* (Lecture Notes in Computer Science). Cham, Switzerland: Springer, 2016, pp. 271–283.

[168] A. C. Yao, "Protocols for secure computations," in *Proc. 23rd Annu. Symp. Found. Comput. Sci. (SFCS)*, Chicago, IL, USA, 1982, pp. 160–164.

[169] (Jul. 2011). *Hashcoin: Blind Bitcoin Transfers, Bitcoin Forum*. Accessed: Nov. 18, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=12751.msg315793#msg315793

[170] M. Blum, "Coin flipping by telephone," in *Proc. Adv. Cryptol. (CRYPTO)*, Santa Barbara, CA, USA, 1981, pp. 11–15.

[171] (Dec. 2011). *Meni Rosenfeld: Using Mixing Transactions to Improve Anonymity, Bitcoin Forum*. Accessed: Feb. 1, 2017. [Online]. Available: https://bitcointalk.org/index.php?topic=54266

[172] E. Z. Yang. (Jul. 2012). *Secure Multiparty Bitcoin Anonymization*. Accessed: Dec. 3, 2016. [Online]. Available: http://blog.ezyang.com/2012/07/secure-multiparty-bitcoin-anonymization

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

[173] K. V. Jónsson, G. Kreitz, and M. Uddin, "Secure multi-party sorting and applications," IACR Cryptol. ePrint Archive, Rep. 2011/122, Mar. 2011. [Online]. Available: https://eprint.iacr.org/2011/122

[174] I. Damgård, M. Geisler, M. Krøigaard, and J. B. Nielsen, "Asynchronous multiparty computation: Theory and implementation," in *Public Key Cryptography—PKC 2009* (LNCS 5443). Heidelberg, Germany: Springer, 2009, pp. 160–179.

[175] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM J. Comput.*, vol. 18, no. 1, pp. 186–208, Feb. 1989.

[176] *Zero Knowledge Contingent Payment, Bitcoin Wiki*. Accessed: Nov. 25, 2016. [Online]. Available: https://en.bitcoin.it/wiki/Zero_Knowledge_Contingent_Payment

[177] *The First Successful Zero-Knowledge Contingent Payment, Bitcoin Core*. Accessed: Nov. 25, 2016. [Online]. Available: https://bitcoincore.org/en/2016/02/26/zero-knowledge-contingent-payments-announcement

[178] W. Banasik, S. Dziembowski, and D. Malinowski, "Efficient Zero-knowledge contingent payments in Cryptocurrencies without scripts," in *Computer Security—ESORICS 2016* (LNCS 9879). Cham, Switzerland: Springer, 2016, pp. 261–280.

[179] Y. Lindell and B. Pinkas, "Secure two-party computation via cut-and-choose oblivious transfer," in *Theory of Cryptography—TCC 2011* (LNCS 6597). Heidelberg, Germany: Springer, 2011, pp. 329–346.

[180] W. Ladd. (Mar. 2012). *Blind Signatures for Bitcoin Transaction Anonymity*. Accessed: Jan. 9, 2017. [Online]. Available: http://wbl.github.io/bitcoinanon.pdf

[181] A. Boldyreva, "Threshold signatures, multisignatures and blind signatures based on the gap-Diffie–Hellman-group signature scheme," in *Public Key Cryptography—PKC 2003* (LNCS 2567). Heidelberg, Germany: Springer, 2002, pp. 31–46.

[182] E. Duffield and K. Hagan. (Mar. 2014). *Darkcoin: Peer-to-Peer Crypto-Currency with Anonymous Blockchain Transactions and An Improved Proof-of-Work System*. Accessed: Dec. 22, 2016. [Online]. Available: https://www.dash.org/wp-content/uploads/2014/09/DarkcoinWhitepaper.pdf

[183] F.-G. Jeng, T.-L. Chen, and T.-S. Chen, "An ECC-based blind signature scheme," *J. Netw.*, vol. 5, no. 8, pp. 921–928, 2010.

[184] E. Duffield and D. Diaz. (Apr. 2014). *Dash: A Privacy-Centric Crypto-Currency*. Accessed: Jan. 5, 2017. [Online]. Available: https://www.dash.org/wp-content/uploads/2015/04/Dash-WhitepaperV1.pdf

[185] E. Duffield. *Dash V13 Evolution Design Overview*. Accessed: Jan. 5, 2017. [Online]. Available: https://www.dash.org/binaries/evo/DashPaper-v13-v1.pdf

[186] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Advances in Cryptology—ASIACRYPT* (LNCS 2248). Heidelberg, Germany: Springer, 2001, pp. 552–565.

[187] N. van Saberhagen. (Oct. 2013). *CryptoNote V 2.0 Whitepaper*. Accessed: Jan. 10, 2017. [Online]. Available: https://cryptonote.org/whitepaper.pdf

[188] E. Fujisaki and K. Suzuki, "Traceable ring signature," in *Public Key Cryptography—PKC 2007* (LNCS 4450). Heidelberg, Germany: Springer, 2007, pp. 181–200.

[189] Bytecoin Team. (Sep. 2015). *Aggregate Addresses in CryptoNote: Towards Efficient Privacy*. Accessed: Jan. 4, 2017. [Online]. Available: https://bytecoin.org/static/files/docs/aggregate-addresses.pdf

[190] "DigitalNote, XDN-project," DigitalNote, White Paper, Jun. 2015, accessed: Jan. 25, 2017. [Online]. Available: http://digitalnote.org/whitepaper.pdf

[191] "The anonymity of darknetspace," DarkNetSpace, White Paper, Oct. 2014, accessed: Jan. 20, 2017. [Online]. Available: http://www.darknetspace.org/files/whitepaper.pdf

[192] (Jun. 2014). *Aeon: [ANN] AEON [2017-03-04: Important Update to 0.9.8.0], Bitcoin Forum*. Accessed: Jan. 27, 2017. [Online]. Available: https://bitcointalk.org/index.php?topic=641696.0

[193] (Aug. 2014). *Monero: [XMR] Monero—A Secure, Private, Untraceable Cryptocurrency, Bitcoin Forum*. Accessed: Nov. 13, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=583449.0

[194] S. Noether and S. Noether. (Sep. 2014). *Monero is Not That Mysterious*. Accessed: Nov. 16, 2016. [Online]. Available: https://lab.getmonero.org/pubs/MRL-0003.pdf

[195] S. Noether. (Jul. 2014). *Review of CryptoNote White Paper*. Accessed: Nov. 16, 2016. [Online]. Available: https://downloads.getmonero.org/whitepaper_review.pdf

[196] S. Noether, S. Noether, and A. Mackenzie. (Sep. 2014). *A Note on Chain Reactions in Traceability in CryptoNote 2.0*. Accessed: Nov. 16, 2016. [Online]. Available: https://lab.getmonero.org/pubs/MRL-0001.pdf

[197] A. Mackenzie, S. Noether, and Monero Core Team. (Jan. 2015). *Improving Obfuscation in the CryptoNote Protocol*. Accessed: Nov. 16, 2016. [Online]. Available: https://lab.getmonero.org/pubs/MRL-0004.pdf

[198] J. Macheta, S. Noether, S. Noether, and J. Smooth. (Sep. 2014). *Counterfeiting Via Merkle Tree Exploits Within Virtual Currencies Employing the CryptoNote Protocol*. Accessed: Nov. 16, 2016. [Online]. Available: https://lab.getmonero.org/pubs/MRL-0002.pdf

[199] G. Maxwell. *Confidential Transactions*. Accessed: Nov. 25, 2016. [Online]. Available: https://people.xiph.org/~greg/confidential_values.txt

[200] G. Maxwell and A. Poelstra. (Jun. 2015). *Borromean Ring Signatures*. Accessed: May 13, 2017. [Online]. Available: https://github.com/Blockstream/borromean_paper/raw/master/borromean_draft_0.01_8c3f9e7.pdf

[201] S. Noether, "Ring confidential transactions," IACR Cryptol. ePrint Archive, Rep. 2015/1098, Nov. 2015. [Online]. Available: https://eprint.iacr.org/2015/1098

[202] S. Noether, A. Mackenzie, and Monero Core Team. (Feb. 2016). *Ring Confidential Transactions*. Accessed: Nov. 16, 2016. [Online]. Available: https://lab.getmonero.org/pubs/MRL-0005.pdf

[203] *Boolberry*. Accessed: Jan. 23, 2017. [Online]. Available: http://boolberry.com/

[204] *Boolberry Solves CryptoNote Issues*, Boolberry, accessed: Jan. 23, 2017. [Online]. Available: http://boolberry.com/files/Boolberry_Reduces_Blockchain_Bloat.pdf

[205] *Boolberry Solves CryptoNote Issues*, Boolberry, accessed: Jan. 23, 2017. [Online]. Available: http://boolberry.com/files/Boolberry_Solves_CryptoNote_Issues.pdf

[206] Rynomster and Tecnovert. *ShadowCash: Zero-Knowledge Anonymous Distributed E-Cash via Traceable Ring Signatures*. Accessed: Jan. 17, 2017. [Online]. Available: http://bravenewcoin.com/assets/Whitepapers/ShadowCash-Zeroknowledge-Anonymous-Distributed-ECash.pdf

[207] *[SDC] ShadowCash |Welcome to the UMBRA, Bitcoin Forum*. Accessed: Dec. 25, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=745352

[208] *Shadow Project*. Accessed: Dec. 26, 2016. [Online]. Available: https://shadowproject.io/en

[209] Hondo |Stealthcoin. (Sep. 2014). *Stealthsend Whitepaper Brief*. Accessed: Jan. 21, 2017. [Online]. Available: https://www.stealthcoin.com/wp-content/uploads/Stealthsend_Whitepaper_brief0914.pdf

[210] N. Chandran, J. Groth, and A. Sahai, "Ring signatures of sub-linear size without random oracles," in *Automata, Languages and Programming* (LNCS 4596). Heidelberg, Germany: Springer, 2007, pp. 423–434.

[211] M. K. Franklin and H. Zhang, "A framework for unique ring signatures," IACR Cryptol. ePrint Archive, Rep. 2012/577, Oct. 2012. [Online]. Available: https://eprint.iacr.org/2012/577

[212] R. Mercer, "Privacy on the blockchain: Unique ring signatures," *arXiv: 1612.01188*, Dec. 2016.

[213] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in *Advances in Cryptology—EUROCRYPT 2003* (LNCS 2656). Heidelberg, Germany: Springer, 2003, pp. 416–432.

[214] A. Saxena, J. Misra, and A. Dhar, "Increasing anonymity in Bitcoin," in *Financial Cryptography and Data Security* (LNCS 8438). Heidelberg, Germany: Springer, 2014, pp. 122–139.

[215] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous distributed E-cash from Bitcoin," in *Proc. IEEE Symp. Security Privacy*, Berkeley, CA, USA, 2013, pp. 397–411.

[216] C. P. Schnorr, "Efficient signature generation for smart cards," *J. Cryptol.*, vol. 4, no. 3, pp. 239–252, 1991.

[217] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," in *Advances in Cryptology—CRYPTO '86* (Lecture Notes in Computer Science), vol. 263. Heidelberg, Germany: Springer, 1986, pp. 186–194.

[218] J. Benaloh and M. de Mare, "One-way accumulators: A decentralized alternative to digital signatures," in *Advances in Cryptology—EUROCRYPT '93* (Lecture Notes in Computer Science), vol. 765. Heidelberg, Germany: Springer, 1994, pp. 274–285.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

[219] C. Garman, M. Green, I. Miers, and A. D. Rubin, "Rational Zero: Economic security for Zerocoin with everlasting anonymity," in *Financial Cryptography and Data Security* (LNCS 8438). Berlin, Germany: Springer, 2014, pp. 140–155.

[220] G. Danezis, C. Fournet, M. Kohlweiss, and B. Parno, "Pinocchio coin: Building Zerocoin from a succinct pairing-based proof system," in *Proc. 1st ACM Workshop Lang. Support Privacy Enhancing Technol. (PETShop)*, Berlin, Germany, 2013, pp. 27–30.

[221] B. Parno, J. Howell, C. Gentry, and M. Raykova, "Pinocchio: Nearly practical verifiable computation," in *Proc. IEEE Symp. Security Privacy*, Berkeley, CA, USA, 2013, pp. 238–252.

[222] E. Androulaki and G. O. Karame, "Hiding transaction amounts and balances in Bitcoin," in *Trust and Trustworthy Computing* (LNCS 8564). Cham, Switzerland: Springer, 2014, pp. 161–178.

[223] E. B. Sasson *et al.*, "Zerocash: Decentralized anonymous payments from Bitcoin," in *Proc. IEEE Symp. Security Privacy*, San Jose, CA, USA, 2014, pp. 459–474.

[224] *Zcash—All Coins Are Created Equal*. Accessed: Jan. 15, 2017. [Online]. Available: https://z.cash/

[225] D. Hopwood, S. Bowe, T. Hornby, and N. Wilcox. *Zcash Protocol Specification*. Accessed: Jan. 20, 2017. [Online]. Available: https://github.com/zcash/zips/blob/master/protocol/protocol.pdf

[226] *Komodo Platform—Making Bitcoin More Anonymous, Secure and Green*. Accessed: Jan. 27, 2017. [Online]. Available: https://komodoplatform.com

[227] (Sep. 2016). *jl777: [ANN][KMD][dPoW] Komodo—Zcash Zero Knowledge Privacy Secured by Bitcoin, Bitcoin Forum*. Accessed: Jan. 27, 2017. [Available]. Online: https://bitcointalk.org/index.php?topic=1605144.0

[228] (Nov. 2016). *Extrabyte: [ANN] Ebitz—237 BTC in 8 days—POS—Zero Knowledge Proof—Deep Web, Bitcoin Forum*. Accessed: Jan. 27, 2017. [Online]. Available: https://bitcointalk.org/index.php?topic=1681965.0

[229] (Aug. 2013). *Gmaxwell: Really Ultimate Blockchain Compression: CoinWitness, Bitcoin Forum*. Accessed: Nov. 22, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=277389.0

[230] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *Proc. IEEE Symp. Security Privacy (SP)*, San Jose, CA, USA, 2016, pp. 839–858.

[231] F. Schuh and D. Larimer. (2015). *Bitshares 2.0: General Overview*. Accessed: Jan. 15, 2017. [Online]. Available: http://docs.bitshares.org/_downloads/bitshares-general.pdf

[232] D. Lukianov. (2015). *Compact Confidential Transactions for Bitcoin*. Accessed: Dec. 19, 2016. [Online]. Available: http://voxelsoft.com/dev/cct.pdf

[233] X. Yi, R. Paulet, and E. Bertino, *Homomorphic Encryption and Applications* (Springer Briefs in Computer Science). Cham, Switzerland: Springer, 2014, p. 27.

[234] (Oct. 2013). *adam3us: Bitcoins With Homomorphic Value (Validatable But Encrypted), Bitcoin Forum*. Accessed: Nov. 25, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=305791.0

[235] *The Elements Project*. Accessed: Jan. 5, 2017. [Online]. Available: https://elementsproject.org

[236] *Blockstream—Technology*. Accessed: Jan. 2, 2017. [Online]. Available: https://blockstream.com/developers

[237] A. Back *et al.* (2014). *Enabling Blockchain Innovations With Pegged Sidechains*. Accessed: Jan. 2, 2017. [Online]. Available: https://blockstream.com/sidechains.pdf

[238] D. Larimer, C. Hoskinson, and S. Larimer. (2013). *BitShares A Peer-to-Peer Polymorphic Digital Asset Exchange*. Accessed: Jan. 15, 2017. [Online]. Available: https://www.weusecoins.com/assets/pdf/library/BitShares%20A%20Peer-to-Peer%20Polymorphic%20Digital%20Asset%20Exchange.pdf

[239] F. Schuh and D. Larimer. (2015). *Bitshares 2.0: Financial Smart Contract Platform*. Accessed: Jan. 15, 2017. [Online]. Available: http://docs.bitshares.org/_downloads/bitshares-financial-platform.pdf

[240] F. Boudot, "Efficient proofs that a committed number lies in an interval," in *Advances in Cryptology—EUROCRYPT 2000* (LNCS 1807). Heidelberg, Germany: Springer, 2000, pp. 431–444.

[241] (Aug. 2016). *Tonych: Hiding Entire Content of On-Chain Transactions, Bitcoin Forum*. Accessed: Dec. 28, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=1574508.0

[242] A. Churyumov. *Byteball: A Decentralized System for Storage and Transfer of Value*. Accessed: Dec. 27, 2016. [Online]. Available: https://byteball.org/Byteball.pdf

[243] (Sep. 2016). *Tonych: BYTEBALL: Totally New Consensus Algorithm + Private Untraceable Payments, Bitcoin Forum*. Accessed: Dec. 27, 2016. [Online]. Available: https://bitcointalk.org/index.php?topic=1608859.0

[244] J. Jogenfors, "Quantum Bitcoin: An anonymous and distributed currency secured by the no-cloning theorem of quantum mechanics," *arXiv preprint arXiv:1604.01383*, Apr. 2016. [Online]. Available: https://arxiv.org/abs/1604.01383

[245] W. K. Wootters and W. H. Zurek, "A single quantum cannot be cloned," *Nature*, vol. 299, pp. 802–803, Oct. 1982.

[246] M. Crosby, Nachiappan, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology beyond Bitcoin," Sutardja Center Entrepreneurship Technol., Berkeley, CA, USA, Rep., Oct. 2015, accessed: Jan. 18, 2017. [Online]. Available: http://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf

[247] A. A. Thu and K. T. Mya, "Implementation of an efficient blind signature scheme," *Int. J. Innov. Manag. Technol.*, vol. 5, no. 6, pp. 443–448, 2014.

[248] I. Bütün and M. Demirer, "A blind digital signature scheme using elliptic curve digital signature algorithm," *Turkish J. Elect. Eng. Comput. Sci.*, vol. 21, no. 4, pp. 945–956, 2013.

[249] A. Küpçü and A. Lysyanskaya, "Usable optimistic fair exchange," in *Topics in Cryptology—CT-RSA 2010* (LNCS 5985). Heidelberg, Germany: Springer, 2010.

[250] C. Orlandi, "Is multiparty computation any good in practice?" in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Prague, Czech Republic, 2011, pp. 5848–5851.

[251] J. Launchbury, D. Archer, T. DuBuisson, and E. Mertens, "Application-scale secure multiparty computation," in *ESOP 2014 Programming Languages and Systems* (LNCS 8410). Heidelberg, Germany: Springer, 2014, pp. 8–26.

[252] S. Bowe, A. Gabizon, and I. Miers, "Scalable multi-party computation for zk-SNARK parameters in the random beacon model," IACR Cryptol. ePrint Archive, Rep. 2017/1050, Nov. 2017. [Online]. Available: https://eprint.iacr.org/2017/1050

[253] S. Bowe. (Sep. 2017). *Cultivating Sapling: Faster zk-SNARKs, ZCash Blog*. Accessed: Nov. 9, 2017. [Online]. Available: https://z.cash/blog/cultivating-sapling-faster-zksnarks.html

[254] K.-A. Shim, "An efficient ring signature scheme from pairings," *Inf. Sci.*, vol. 300, pp. 63–69, Apr. 2015.

[255] J. Sen, "Homomorphic encryption: Theory & applications," *arXiv preprint arXiv: 1305.5886*, May 2013. [Online]. Available: https://arxiv.org/abs/1305.5886

[256] Y. Hu, "Improving the efficiency of homomorphic encryption schemes," Ph.D. dissertation, Dept. Elect. Comput. Eng., Worchester Polytechnic Inst., Worcester, MA, USA, 2013, accessed: Nov. 14, 2017. [Online]. Available: https://web.wpi.edu/Pubs/ETD/Available/etd-042513-154859/unrestricted/YHu.pdf

[257] C. Jost, H. Lam, A. Maximov, and B. J. M. Smeets, "Encryption performance improvements of the Paillier cryptosystem," IACR Cryptol. ePrint Archive, Rep. 2015/864, Sep. 2015. [Online]. Available: https://eprint.iacr.org/2015/864

[258] C. Decker and R. Wattenhofer, "A fast and scalable payment network with Bitcoin duplex micropayment channels," in *SSS 2015 Stabilization, Safety, and Security of Distributed Systems* (LNCS 9212). Cham, Switzerland: Springer, 2015, pp. 3–18.

[259] M. Green and I. Miers, "Bolt: Anonymous payment channels for decentralized currencies," IACR Cryptol. ePrint Archive, Rep. 2016/701, Aug. 2016. [Online]. Available: https://eprint.iacr.org/2016/701

[260] A. Kumar, C. Fischer, S. Tople, and P. Saxena, "A traceability analysis of Monero's blockchain," in *Proc. Eur. Symp. Res. Comput. Security (ESORICS)*, Oslo, Norway, 2017, pp. 153–173.

[261] T. Ruffing and G. Malavolta, "Switch commitments: A safety switch for confidential transactions," IACR Cryptol. ePrint Archive, Rep. 2017/237, 2017. [Online]. Available: https://eprint.iacr.org/2017/237

[262] C. Yuan, M.-X. Xu, and X.-M. Si, "Research on a new signature scheme on blockchain," *Security Commun. Netw.*, vol. 2017, p. 10, Aug. 2017.

[263] M. Campanelli, R. Gennaro, S. Goldfeder, and L. Nizzardo, "Zero-knowledge contingent payments revisited: Attacks and payments for services," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, Dallas, TX, USA, 2017, pp. 229–243.

[264] *Zcash—Fixing Vulnerabilities in the Zcash Protocol*. Accessed: Nov. 22, 2017. [Online]. Available: https://z.cash/blog/fixing-zcash-vulns.html

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.

[265] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-NG: A scalable blockchain protocol," in *Proc. 13th Usenix Conf. Netw. Syst. Design Implement.*, Santa Clara, CA, USA, 2016, pp. 45–59.
[266] G. O. Karame, "On the security and scalability of Bitcoin's blockchain," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, Vienna, Austria, 2016, pp. 1861–1862.
[267] J. Herrera-Joancomartí and C. Pérez-Solà, "Privacy in Bitcoin transactions: New challenges from blockchain scalability solutions," in *Modeling Decisions for Artificial Intelligence* (LNCS 9880). Cham, Switzerland: Springer, 2016, pp. 26–44.
[268] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Scalable zero knowledge via cycles of elliptic curves," *Algorithmica*, vol. 79, no. 4, pp. 1102–1160, 2017.
[269] Y. Zhang, Y. Long, Z. Liu, Z. Liu, and D. Gu, "Z-channel: Scalable and efficient scheme in Zerocash," IACR Cryptol. ePrint Archive, Rep. 2017/684, 2017. [Online]. Available: https://eprint.iacr.org/2017/684
[270] S. Sun, M. H. Au, J. K. Liu, and T. H. Yuen, "RingCT 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency Monero," in *Computer Security—ESORICS 2017* (LNCS 10493). Cham, Switzerland: Springer, 2017, pp. 456–474.
[271] *Reddit: Question: How to Trust Monero?* Accessed: Oct. 23, 2017. [Online]. Available: https://www.reddit.com/r/Monero/comments/6zyndw/question_how_to_trust_monero/
[272] *Reddit: Is it Possible to Audit the Number of Coins in Monero?* Accessed: Oct. 23, 2017. [Online]. Available: https://www.reddit.com/r/Monero/comments/6cu2qm/is_it_possible_to_audit_the_number_of_coins_in/

Merve Can Kus Khalilov received B.S. and M.S. degrees in computer engineering from Istanbul Technical University, Istanbul, Turkey, in 2006 and 2008, respectively. She is currently pursuing the Ph.D. degree computer science and engineering with the Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey. She has been with Research and Development Center, Kuveyt Turk Participation Bank, Çayirova, Turkey, since 2010. She served as Enterprise Architect, and is currently FinTech Research and Development Team Leader. Her research interests include information security, privacy, digital cash mechanisms, and financial technologies.

Albert Levi received B.S., M.S., and Ph.D. degrees in computer engineering from Boğaziçi University, Istanbul, Turkey, in 1991, 1993, and 1999, respectively. He served as a Visiting Faculty Member with the Department of Electrical and Computer Engineering, Oregon State University, OR, from 1999 to 2002, where he was also a Post-Doctoral Research Associate with the Information Security Laboratory. Since 2002, he has been a Faculty Member of computer science and engineering with the Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, where he is a Founding Co-Director of the Cryptography and Information Security Group. He has been promoted to an Associate Professor in 2008 and to a Full Professor in 2015. His research interests include computer and network security with emphasis on mobile and wireless system security, public key infrastructures, privacy, and application layer security protocols. He has served in the program committees of various international conferences. He also served as the General and Program Co-Chair of ISCIS 2006, the General Chair of SecureComm 2008, the Technical Program Co-Chair of NTMS 2009, the Publicity Chair of GameSec 2010, and the Program Co-Chair of ISCIS 2011. He is the Editorial Board Member of the *Computer Journal* published by Oxford University Press and *Computer Networks* published by Elsevier.

Authorized licensed use limited to: Peking University. Downloaded on June 29,2023 at 01:44:49 UTC from IEEE Xplore. Restrictions apply.