Towards Anonymous yet Accountable Authentication for Public Wi-Fi Hotspot Access With Permissionless Blockchains

Yukun Niu, Lingbo Wei, *Member, IEEE*, Chi Zhang, *Member, IEEE*, Jianqing Liu, *Member, IEEE*, and Yuguang Fang, *Fellow, IEEE*

***Abstract*—Anonymous yet accountable authentication can protect users' privacy and security and prevent users from misbehaving when they access public Wi-Fi hotspots. However, most existing privacy-enhanced authentication schemes either do not meet the accountability requirements in public Wi-Fi hotspot access or they are inherently dependent on trusted third parties, and therefore are undeployable in practical settings. In this paper, we design and implement an access authentication scheme to simultaneously and efficiently provide anonymity and accountability without relying on any trusted third party by utilizing a permissionless blockchain (e.g., Bitcoin or Ethereum) and Intel SGX. Inspired by the recent progress on Bitcoin techniques such as Colored Coins, we utilize the unmodified Bitcoin blockchain as the powerful platform to manage access credentials without introducing any trusted third party. We leverage SGX-based mixer to allow users to anonymously exchange their access credentials and design the verification path of access credentials to support blacklisting misbehaving access credentials without compromising users' anonymity. By integrating with the anti-double-spending property of the Bitcoin blockchain, our scheme can simultaneously provide users' accountability and anonymity without involving any trusted third party. Finally, we demonstrate that our proposed scheme is compatible with the current Bitcoin system or other permissionless blockchains, and is highly effective and practical for public Wi-Fi hotspot access control systems.**

***Index Terms*—Blockchains, authentication.**

Manuscript received 6 March 2022; revised 19 July 2022; accepted 29 September 2022. Date of publication 1 November 2022; date of current version 14 March 2023. This work was supported in part by the Natural Science Foundation of China under Grants 62072426, 61871362, U19B2023, and U21B2026, in part by the Key Research and Development Program of Anhui under Grant 202104b11020029 and in part by the Innovation and Application Demonstration of Social Management Information Technology of Hainan Free Trade Port under Grant ZDKJ2019008. The work by Jianqing Liu was supported in part by the National Science Foundation under Grants ECCS-2047484 and CNS-2211214. The review of this article was coordinated by Prof. Yu Cheng. An earlier version of this paper was presented in part at the 2017 IEEE/CIC International Conference on Communications in China (IEEE/CIC ICCC) [DOI: 10.1109/ICChina.2017.8330337]. *(Corresponding author: Chi Zhang.)*

Yukun Niu and Chi Zhang are with the School of Information Science and Technology, University of Science and Technology of China, Hefei 230027, China (e-mail: xiaoniu@mail.ustc.edu.cn; chizhang@ustc.edu.cn).

Lingbo Wei is with the School of Information Science and Technology, University of Science and Technology of China, Hefei 230027, China, and also with the Institute of Dataspace, Hefei Comprehensive National Science Center, Hefei 230027, China (e-mail: lingbowei@ustc.edu.cn).

Jianqing Liu is with the Department of Computer Science, North Carolina State University, Raleigh, NC 27606 USA (e-mail: jliu96@ncsu.edu).

Yuguang Fang is with the Department of Computer Science, City University of Hong Kong, Hong Kong, SAR, China (e-mail: my.fang@cityu.edu.hk).

Digital Object Identifier 10.1109/TVT.2022.3218528

I. Introduction

OPEN hotspots hosted by untrusted access points (APs) have been widely deployed to provide ubiquitous Wi-Fi connectivity and support traffic offloading to Wi-Fi hotspots as a solution to traffic congestion for mobile service providers. Due to the open and distributed nature of Wi-Fi connectivity, it is important to enforce hotspot access control through user authentication schemes to cope with both free riders and malicious attackers. In current authentication scheme like WPA2, dishonest network service providers or untrusted APs might threaten users' privacy by tracking which APs a user connects to and when, and then revealing user's mobility patterns and other sensitive information [2].

Anonymous authentication is a promising solution to provide secure, privacy-preserving authentication and access control in such scenarios [3]. Anonymous authentication schemes can hide user's real identity from both AP operators and network service providers, and make different accesses performed by the same user unlinkable. At the same time, mutual authentication can still be securely implemented to allow only legitimate users to connect to legal hotspot APs.

However, unconditional anonymity introduces a new problem for a network service provider. Since all anonymous users look alike, there is no way for the network service provider to hold individual misbehaving anonymous users accountable for their actions [4]. In public Wi-Fi hotspot access, misbehaving users may share their access credentials [5] to free riders or launching a selfish attack [6] to increase their share of bandwidth at the expense of other users. To regulate user behaviors and protect public Wi-Fi hotspots from being abused and attacked, an anonymous authentication scheme should empower a network service provider with the ability to revoke the Wi-Fi hotspot access from abusive anonymous users.

While anonymous authentication protocols exist for wireless access control, only few of them can provide accountability against misbehaving users. Funabiki et al. in [7] utilized a group manager to manage the personal information of every user and deanonymize the group signature to trace a misbehaving user, where the group manager is a trusted third party (TTP). In [8], [9], it was proposed to separate management functionalities between the group manager and the network operator to achieve anonymity and accountability for wireless access networks, and thus they need to assume that the group manager does not collude with the network operator. In this paper, we treat these non-colluded parties as a TTP because they can deanonymize a user without the user's awareness if they collude with each other.

Blacklistable anonymous credentials can be applied in public Wi-Fi hotspot access to achieve anonymous yet accountable authentication without relying on any TTP. Blacklistable anonymous credential schemes allow a service provider to block misbehaving users without involving any TTP, and so these schemes can prevent users from launching selfish attacks. The blacklistable anonymous credential was first proposed in [4], i.e., BLAC. When a user anonymously authenticates to his/her service provider, he/she first proves that he/she is not on the service provider's blacklist. Therefore, the computational complexity of the authentication is linear with the size of the blacklist. The follow-up work attempted to reduce computational complexity of the authentication or support reputation. Among them, FARB [10] and the scheme in [11] had constant computational complexity, both for the service provider and the user. In comparison with FARB, the scheme in [11] had the advantage in the redeem protocol at the cost of the authentication performance on the user side. However, all the blacklistable anonymous credential schemes cannot prevent users from sharing their credentials.

In recent years, blockchain-based identity management has attracted massive attention both from academy [12], [13], [14] and industry communities, and many related schemes are proposed, such as ShoCard, BitID, IDchainZ, Civic, Sovrin, uPort, and Onename [15], [16]. Blockchain-based identity management schemes eliminate the need for a central authority to control and manage users' identities. Moreover, blockchain-based identity management schemes can be utilized to achieve anonymous authentication with privacy-preserving methods, such as zero-knowledge proof [12], the modified Coconut blind signature [13], and the Shamir's secret sharing [14]. Additionally, some anonymous authentication schemes utilizing blockchain are proposed for VANETs [17], [18], [19] or Space-Ground Integrated Networks [20]. Unfortunately, these schemes either utilize permissioned blockchain or support users' accountability with a TTP.

Although permissioned blockchains can be highly customized and therefore can support specific applications more efficiently than permissionless blockchains, the latter are still the most suitable platform for public Wi-Fi hotspot access control. First of all, permissionless blockchains are more secure than permissioned ones. The security of (permissioned or permissionless) blockchains is not proven in theory, but tested in practice [21]. The mainstream permissionless blockchains, especially the Bitcoin blockchain, have survived the rigorous test of time, but it is not the case for permissioned blockchains. Secondly, permissionless blockchains are open to all users and therefore can be utilized by any operator to provide public Wi-Fi hotspot access. At the same time, users will not be bothered by the so-called "vendor lock-in effect": they can move from one vendor to another without substantial switching costs [22]. Thirdly, permissionless blockchains are existing infrastructures that can be utilized by the operators with small or no fixed costs [23]. However, for a permissioned solution, there is no existing infrastructure; therefore, the operator must pay a large fixed cost to build and maintain a secure and decentralized infrastructure (the underlying permissioned blockchain) at the very beginning.

In this paper, we propose an anonymous and accountable access authentication scheme for Wi-Fi hotspot access without using any TTP. Inspired by Colored Coins [24], our proposed scheme utilizes the unmodified Bitcoin blockchain as the platform to manage users' access credentials. We associate user's access right with its Bitcoin address, which can be utilized as a credential to access public Wi-Fi hotspots. Moreover, our proposed scheme leverages a SGX-based mixer to allow users to exchange their access credentials anonymously and securely. Verification path is constructed to ensure that one old credential only makes one new credential valid. Combining with anti-double-spending property of the Bitcoin blockchain, the number of valid credentials owned by users will not be larger than the number of original credentials obtained in user registration phase. Verification path can also be utilized to blacklist misbehaving access credentials without compromising users' anonymity. Note that our proposed scheme can also be directly applied into other permissionless blockchains, such as Bitcoin Cash or Ethereum.

We have proposed the initial design of our scheme in [1], where the security properties and feasibility were not fully investigated. As a result, we provide detailed security and efficiency analysis in this paper to show that our scheme is a practically viable solution to public Wi-Fi hotspot access. Moreover, we utilize Intel SGX-based mixing protocol in this paper, instead of CoinShuffle [25]. Combining with payment channel technique, our scheme supports continuous access credential exchange without any change addresses,[1] which can be a side-channel utilized to deanonymize user's access credentials if a user's change addresses are linked together.

The remainder of this paper is organized as follows. Section II defines the notations used in the paper and gives a short description of the elliptic curve integrated encryption scheme (ECIES). In Section III, we elaborate the system model, basic assumptions, and design goals of our scheme. The access credential management scheme based on the Bitcoin blockchain is introduced in Section IV and the anonymous yet accountable access credential management scheme is described in Section V. Section VI offers the security analysis and Section VII conducts the efficiency analysis and performance evaluation. We conclude this paper in Section VIII.

II. Notations and Preliminaries

A. Notations

- U, AS, M, tx: abbreviations for user, authentication server, secure enclave in the SGX-based mixer, and transaction, respectively;
- $txid$: the identifier of a transaction. In fact, $txid$ is the hash of a transaction;

[1] In Bitcoin, a change address is utilized to transfer the remainder of the payment back to the payer because bitcoins cannot be spent partially.

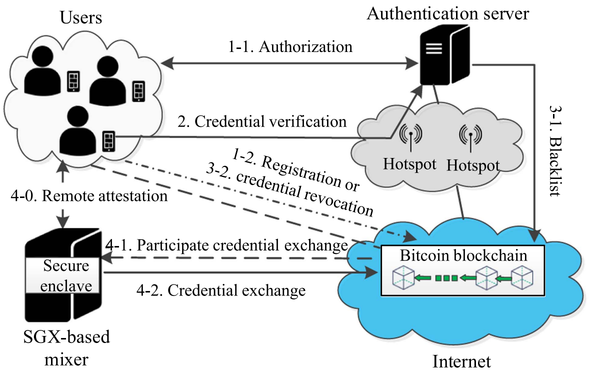


Fig. 1. System model and scheme overview.

- AC_x: the access credential of entity x;
- pk_x: the public key of entity x;
- sk_x: the private key of entity x;
- $k_{i,j}$: a symmetric key negotiated by parties i and j utilizing ECIES;
- $x||y$: the concatenation of x and y;
- $\mathrm{Sig}(sk_x, m)$: the signature of message m using x's private key;
- $\mathrm{Enc}(k, m)$: the encryption of message m using symmetric key k;
- $\mathrm{Dec}(k, c)$: the decryption of ciphertext c using symmetric key k;
- $\mathrm{H}(m)$: the hash of message m.

B. ECIES

ECIES (Elliptic Curve Integrated Encryption Scheme) is a hybrid encryption scheme which combines the convenience of a public-key cryptosystem with the efficiency of a symmetric-key cryptosystem [26]. The security of ECIES is based on the hardness of the discrete logarithm problem on elliptic curves. ECIES allows two parties to generate a shared key which can be utilized for designated symmetric encryption algorithm (such as AES). Given an elliptic group $E_p(a, b)$ and its generator point G. The private key of a user is a random number $d \leftarrow Z_q^*$ and the corresponding public key is $Q = d \cdot G$. Assume that Alice and Bob have private key d_1 and d_2 with corresponding public keys $Q_1 = d_1 \cdot G$ and $Q_2 = d_2 \cdot G$, respectively. The key agreement protocol here is a natural generalization of the original Diffie-Hellman key exchange scheme:

- Alice computes $(k_{\mathrm{MAC}}, k_{\mathrm{ENC}}) = \mathrm{KDF}(d_1 \cdot Q_2 || para)$. The key k_{MAC} is used to encrypt MAC of a message and k_{ENC} is used to encrypt the message. The function $\mathrm{KDF}(\cdot)$ is a key derivation function which produces a set of keys from keying material and some optional parameters $para$, such as the binary representation of the Alice's public key Q_1;
- Bob computes $(k_{\mathrm{MAC}}, k_{\mathrm{ENC}}) = \mathrm{KDF}(d_2 \cdot Q_1 || para)$ with the same optional parameters that Alice used.

Since $d_1 \cdot Q_2 = d_1 \cdot d_2 \cdot G = d_2 \cdot d_1 \cdot G = d_2 \cdot Q_1$, Alice and Bob will generate the same $(k_{\mathrm{MAC}}, k_{\mathrm{ENC}})$.

To encrypt a message m and send it to Bob, Alice performs the following steps:

- Alice generates $(k_{\mathrm{MAC}}, k_{\mathrm{ENC}})$;
- Alice computes ciphertext $c = \mathrm{Enc}(k_{\mathrm{ENC}}, m)$ and $tag = \mathrm{Enc}(k_{\mathrm{MAC}}, \mathrm{H}(m))$ separately;
- Alice sends the message $(Q_1||tag||c)$ to Bob.

Upon receiving message $(Q_1||tag||c)$, Bob performs the following steps:

- Bob generates $(k_{\mathrm{MAC}}, k_{\mathrm{ENC}})$;
- Bob computes $m' = \mathrm{Dec}(k_{\mathrm{ENC}}, c)$ and verifies whether $\mathrm{H}(m') \equiv \mathrm{Dec}(k_{\mathrm{MAC}}, tag)$. If so, $m' = m$.

III. Models and Design Goals

A. System Model and Adversary Model

A general authentication scenario in our scheme involves five parties: a user, a visited hotspot, a service provider's authentication server (AS), the blockchain,[2] and a SGX-based mixer. As shown in Fig. 1, a network service provider (NSP) offers network access to users through a set of hotspots. Generally speaking, an NSP deploys hotspots in public areas, such as parks, airports, and bus/train stations. A user can subscribe to the network service at any time by contacting NSP. After subscription process, the user utilizes his/her mobile device to connect to the visited hotspot and performs mutual authentication among the user, the visited hotspot, and AS. On the one hand, the visited hotspot ensures that the user is a legitimate user who subscribes the network service. On the other hand, the user ensures the visited hotspot is a legitimate AP controlled by NSP, instead of a rogue AP. After mutual authentication, the user can access the Internet through the visited hotspot.

To support mutual authentication, the user leverages the blockchain to manage and determine ownership of access credentials, which can be utilized to prove that the user is a legitimate user. To achieve anonymous authentication, some users can exchange their access credentials by utilizing an SGX-based mixer and then obtain the same number of new anonymous access credentials. An SGX-based mixer is an untrusted entity and consists of two components: an untrusted server and a trusted part, called a secure enclave. When receiving at least k participation transactions, the secure enclave creates a credential exchange transaction and broadcasts the credential exchange transaction into the blockchain network. When the credential exchange transaction is included in the blockchain, the users obtain their new anonymous access credentials and their old access credentials are invalid.

We focus on two types of adversary models: abuse and privacy. On the one hand, some users attempt to perform illegitimate activities using valid credentials (misbehaving), such as launching a selfish attack or sharing their access credentials. The selfish attack can increase misbehaving user's share of bandwidth at the cost of other users. Sharing access credential makes it possible that the misbehaving users or free riders access Internet through the hotspots controlled by NSP. On the other hand, AS, some colluded users, and the SGX-based mixer attempt to link credentials to users' identities or other credentials held by the

[2] The blockchain mentioned in this paper all refers to the Bitcoin blockchain without further clarification.

same user. Notice that AS may abuse accountability mechanism to link a user's credentials if this abuse can deanonymize the user without user's awareness.

B. Design Goals

In light of the above adversary models, the following security requirements are essential to ensure that an authentication scheme for public Wi-Fi hotspot access is anonymous and accountable:

1) Security: The scheme achieves explicit mutual authentication between users and hotspots. On the one hand, the scheme can prohibit unauthorized users from accessing Internet through a visited hotspot. On the other hand, the scheme can prevent rogue hotspots from launching phishing attacks.
2) Anonymity and unlinkability: The scheme enables unilateral anonymous authentication between users and hotspots. That is, no one can link access credentials to users' identities or other access credentials held by the same user.
3) Mutual accountability: On the one hand, AS should be able to prevent misbehaving credential holders from accessing public Wi-Fi hotspots. Thus, the scheme allows dynamic credential revocation. On the other hand, AS cannot abuse accountability power to deanonymize users without users' awareness.
4) Efficiency: Mutual authentication should have low communication and computation overhead.
5) No need of a trusted third party: The scheme does not need the existence of a trusted third party.

IV. Access Credential Management Utilizing the Bitcoin Blockchain

In this section, for better illustration, we describe a simple basic access credential management scheme utilizing the Bitcoin blockchain[3] yet without guaranteeing user's anonymity. It involves a user, a visited hotspot, AS, and the blockchain. Since it has a limited privacy guarantee, we present how to leverage it to construct anonymous yet accountable access credential management scheme to provide the desired privacy properties without compromising user's accountability elaborated in the next section.

It is worth noting that all the credential operations will be realized by creating or reading the Bitcoin transactions. We define different transaction formats for different credential operations in the following sections. All transactions created by our system contain an OP_RETURN output, also called marker output, which can be utilized to distinguish our system transactions from other transactions, and identify transaction type. Moreover, we can embed data in a marker output to realize user registration and blacklist update building blocks. The format of a marker output is OP_RETURN⟨data⟩, where ⟨data⟩ is defined as follows: ⟨identifier⟩⟨version⟩⟨tx_type⟩⟨payload⟩.

A. User Registration

When a user wants to join the system for the first time, it performs user registration to obtain an access credential. The user registration can be divided into two phases: authorization and access credential publishing. In the first phase, AS validates the user's real identity and deposit transaction, and then authorizes the user via a digital signature on a registration transaction. In the second phase, the user completes the registration transaction and then broadcasts the deposit transaction and the registration transaction into the Bitcoin network. When both transactions are included in the blockchain, the user registration is completed. The user registration is described in more detail below.

The deposit transaction has three special outputs: a registration output, a deposit output, and a marker output, which follow this order in the deposit transaction. The registration output is a P2PKH output, which will be spent by an input in the registration transaction. In the P2PKH output, the Bitcoin address is controlled by the user. The deposit output is a 2-of-2 multisig output, in which one Bitcoin address is controlled by the user and the other one is controlled by AS. The payload of the marker output is NULL.

The input of the registration transaction spends the registration output in the deposit transaction. The registration transaction contains two special outputs: an access credential output and the marker output. The access credential output is a P2PKH output, in which AC_i is the hash of the public key. The authorization information is a digital signature, $\mathrm{Sig}(sk_{\mathrm{AS}}, \mathrm{H}(reg\_tx))$, on the registration transaction signed by AS, which is the payload of the marker output.

B. Credential Verification

When a user connects to a visited hotspot, the user needs to prove that it does hold a valid access credential. To do so, the user should prove i) it owns the access credential, i.e. the ownership verification, and ii) the access credential is valid, i.e. the validity verification. The user can utilize a digital signature as the ownership proof to show his/her ownership of the access credential because the access credential is the hash of a public key. To prove a credential is valid, the user just sends the access credential to AS. The AS then validates the access credential with the blockchain and the blacklist introduced in Section IV-C. Only when the following conditions are met can AS ensure that the access credential is valid.

- The access credential is not in the blacklist;
- The access credential is in a registration transaction;
- The registration transaction is included in the blockchain;
- The authorization information in the registration transaction is valid. That is, the digital signature signed on the registration transaction $\mathrm{Sig}(sk_{\mathrm{AS}}, \mathrm{H}(reg\_tx))$ is valid.

C. Credential Revocation

Credential revocation operations are involved in two cases: AS punishes misbehaving users, or a user leaves the system. In the former case, AS revokes an access credential via the

[3] Note that our scheme can be easily extended to other permissionless blockchains (such as Bitcoin Cash or Ethereum).

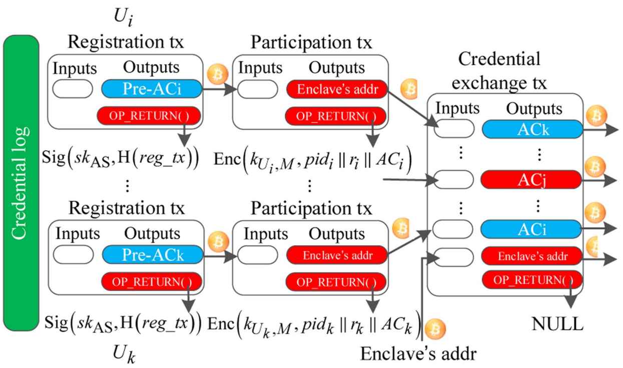


Fig. 2. Credential exchange.

blacklist update. In the latter case, a user needs to revoke its access credential and withdraw its deposit.

When AS detects a misbehaving user, it adds the user's access credential to a blacklist. Then, the blacklisted access credential becomes invalid. AS can update the state of the blacklist via a blacklist update transaction. For example, the blacklist can be organized as a Merkle tree. The leaf of the Merkle tree is a blacklisted access credential. The root of the Merkle tree is the commitment of the blacklist and will be embedded in the payload field of the marker output.

When a user wants to leave the system, it revokes its access credential and requests AS to withdraw its deposit. For a misbehaving user, its access credential has already been revoked by the blacklist, and so it just needs to withdraw its deposit. The withdraw transaction of a misbehaving user will transfer some deposit to AS's Bitcoin address and transfer the others to the user's Bitcoin address. For an honest user, it first generates a credential revocation transaction to revoke its access credential. Then, its withdraw transaction will transfer all deposit to the user's Bitcoin address.

V. Anonymous Yet Accountable Access Credential Management

A. Credential Exchange

Our credential exchange protocol is divided into four phases: remote attestation, participation via blockchain, verification of access credential, and creation of credential exchange transaction.

1) Remote Attestation: Before participating a credential exchange protocol, a user and the secure enclave in an SGX-based mixer perform remote attestation. After the remote attestation, the user and the secure enclave establish a secure channel[4] between them, and then the user obtains the secure enclave's public key pk_M through the secure channel.

Participation via blockchain: A user participates in a credential exchange protocol by submitting a participation transaction to the secure enclave's address. As shown in Fig. 2, the participation transaction spends the output containing the old access credential and contains two outputs. One output is a P2PKH output, which contains the secure enclave's address. Another output is a marker output, which embeds a new access credential and some information for credential maintenance, which will be described in Section V-D, encrypted by a shared key $k_{U_i,M}$. The shared key $k_{U_i,M}$ is generated with the user's secret key sk_{U_i}, corresponding to the old access credential, and the secure enclave's public key pk_M, as described in Section II-B. Then, the user broadcasts the participation transaction in the Bitcoin network.

Verification of access credential: After the participation transaction included in the blockchain, the secure enclave verifies the old access credential as described in the Section V-B. If the old access credential is valid from the perspective of the secure enclave, the secure enclave utilizes the shared key $k_{U_i,M}$ to decrypt the payload of the marker output and obtains the user's new access credential.

Creation of credential exchange transaction: After receiving at least k valid participation requests, the secure enclave creates a credential exchange transaction. The users' new access credentials will be put in the P2PKH outputs. First, the secure enclave shuffles the order of these P2PKH outputs. Then, the secure enclave generates the inputs spending the outputs in the participation transactions. For k participants, the credential exchange transaction contains $k + 1$ inputs and $k + 2$ outputs. Every participant contributes one input, referring to the old access credential, and one output, containing the new access credential. The secure enclave contributes one input to pay the transaction fees and one output to receive the change. Another output is the marker output, whose payload can be NULL. In this way, the value of the outputs containing the old access credentials and the new access credentials are the same. Moreover, the mapping between inputs and outputs in a credential exchange transaction is randomized so that the relationship between old credentials and new credentials cannot be inferred by both outsiders and participants. Thus, we can achieve k-anonymity in the credential exchange protocol.

Our scheme can support off-chain credential exchange in a way similar to the payment channel techniques designed for permissionless blockchains. To do so, the user needs convert his/her on-chain access credential to an off-chain access credential. The user firstly constructs a commitment transaction, whose input contains the on-chain access credential and output contains a secure enclave's address. When this transaction is included on the Bitcoin blockchain, the user's on-chain access right is locked by the secure enclave's address. Then, the user can utilize this locked on-chain access credential as its off-chain access credential with the permission of the secure enclave. When users want to exchange access credentials in an off-chain fashion, each of them can send a participation request, containing the user's original off-chain access credential and a new off-chain access credential, to the secure enclave. After receiving at least k valid participation requests, the secure enclave constructs an off-chain credential exchange transaction, whose inputs contain the old off-chain access credentials and outputs contain the new off-chain access credentials specified by the participation requests. Note that off-chain transactions introduced above will

[4] The user and the secure enclave share a session key and they encrypt their communication messages by utilizing the session key.

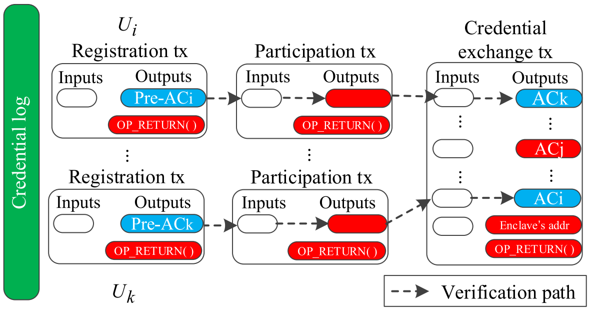


Fig. 3. Verification path.

not be accepted by the blockchain, because the inputs of these transactions have already been spent. But the AS can verify and accept these transactions based on the signature of the secure enclave. If a user wants to convert his/her off-chain access credential back to an on-chain access credential, he/she only needs send a conversion request to the secure enclave. Then, the secure enclave constructs a conversion transaction, whose input spends the output of the user's commitment transaction and output contains the user's new on-chain access credential. When this transaction is included on the Bitcoin blockchain, the user gets his/her on-chain access right again.

B. Advanced Credential Verification

Due to the introducing of credential exchange protocol mentioned above, the credential verification process should be modified accordingly. On the one hand, the new anonymous access credential is not in the registration transaction. On the other hand, we should distinguish the new access credential from old access credentials.

Verification path is proposed to construct a proof to guarantee that the new access credential is referred to a registration transaction. Note that, the number of old access credentials is the same as that of new access credentials in the credential exchange protocol. In the blockchain, a transaction's input contains a reference to a previous transaction's output. Thus, the input can be linked to the previous output. If we define a mapping between the inputs and the outputs in the same transaction, a path is constructed between an access credential and a registration transaction. To realize this idea, the input i is mapped to the output i, i.e., new access credential in the output i can use the registration transaction of input i to prove its validity. Note that the access credential and the registration may be held by different users. As shown in the Fig. 3, user U_i's new access credential AC_i is linked to the registration transaction of the user U_k.

We utilize the UTXO to distinguish the new access credential from old access credentials. If an access credential is not used to participate in a credential exchange protocol, the output containing the access credential should be an UTXO. If the participation transaction, which spends the output containing the access credential, is included in the blockchain, the output of the participation transaction should be an UTXO. Otherwise, the access credential is an old access credential.

To accelerate the speed of credential verification in the mutual authentication or credential exchange protocol, AS and the SGX-based mixer can maintain an access credential set, similar to the UTXO set in the blockchain. The access credential set will be updated when the new block contains transactions with the marker output or the blacklist is updated. In this way, the user just sends its access credential, the corresponding public key, and the corresponding ownership proof to AS in the mutual authentication. After the ownership verification, AS just queries the access credential set to check whether the access credential is valid.

C. Advanced Credential Revocation

The abuse of an access credential may be detected when the credential exchange transaction serving the access credential is broadcasted in the Bitcoin network or included in the blockchain. In this case, blacklisting the access credential is not enough because the misbehaving user will obtain or has already obtained a new anonymous access credential.

To solve this problem, AS adds the access credential into the blacklist, and broadcasts blacklist request to the SGX-based mixer. If the SGX-based mixer mixing the blacklisted access credentials receives the blacklist request, the secure enclave searches for its local mixing log and sends the misbehaving user's new access credential to AS. Then, AS adds the new access credential to the blacklist. To prevent AS from abusing blacklist, the secure enclave will monitor the blacklist update transaction. If AS did not blacklist the access credential provided by the secure enclave before a deadline, such as waiting 1,008 blocks (about 7 days), the secure enclave publishes the misbehavior of AS in the blockchain.

However, the SGX-based mixer mixing the blacklisted access credentials may have been shut down or left the system. In this case, AS blacklists the credential exchange transactions referring to the blacklisted access credentials. The new access credentials in the credential exchange transaction will be at the suspicious state. The user holding an access credential at the suspicious state needs to provide its corresponding old access credential in the mutual authentication to prove that they are not a misbehaving user. The honest user can re-register with the suspicious access credential to obtain a new valid access credential.

D. Credential Maintenance

To achieve k-anonymity, all values of the outputs containing the old and new access credentials should be the same. However, all transactions in the blockchain have the transaction fees. Credential maintenance protocol tries to pay the transaction fees by the secure enclave in the SGX-based mixer. The basic idea is to build a payment channel between a user and the secure enclave, so the transaction fees can be paid by the secure enclave and the user pays for the secure enclave through the payment channel. Suppose that user U_i pays transaction fee f_1 for its participation transaction and the transaction fee of the credential exchange transaction containing k participants is f_2; then the secure enclave transfers $f_1 + \frac{f_2}{k}$ from the user's balance to the secure enclave's balance in the payment channel between them.

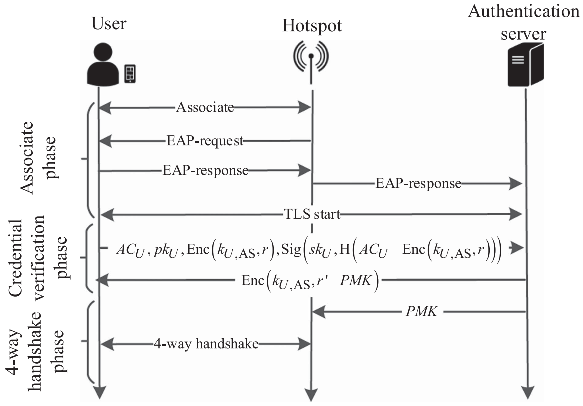

User
Hotspot
Authentication server
Associate phase
Credential verification phase
4-way handshake phase
Associate
EAP-request
EAP-response
EAP-response
TLS start
$AC_U, pk_U, \mathrm{Enc}(k_{U,\mathrm{AS}}, r), \mathrm{Sig}(sk_U, \mathrm{H}(AC_U \quad \mathrm{Enc}(k_{U,\mathrm{AS}}, r)))$
$\mathrm{Enc}(k_{U,\mathrm{AS}}, r' \quad PMK)$
PMK
4-way handshake

Fig. 4. Mutual authentication.

User U_i and secure enclave M establish a payment channel after the remote attestation. User U_i obtains the secure enclave's address through the secure channel between them. Then, the user generates a payment channel establishment transaction. Through the payment channel establishment transaction, the user transfers some coins to the address owned by the secure enclave. After that, the secure enclave defines an identifier number pid_i for the payment channel and generates a random number r_i as the shared secret between them. Then, the secure enclave sends pid_i and r_i to the user.

In the participation transaction, user U_i authorizes the secure enclave to update its balance after generating the credential exchange transaction. To do so, user U_i should embed pid_i and r_i encrypted by the shared key $k_{U_i,M}$ in the participation transaction. In this way, the payload of the marker output in the participation transaction should be $\mathrm{Enc}(k_{U_i,M}, pid_i||r_i||AC_i)$.

E. Mutual Authentication

We can utilize our anonymous yet accountable access credential management scheme as an extension to the Protected EAP to achieve the mutual authentication. In this way, the user and AS can establish a TLS tunnel in the common Wi-Fi WPA-Enterprise framework. Then, the user and AS can perform credential verification through the TLS tunnel. After that, AS shares a pairwise master key *PMK* with the user and the visited hotspot, and so the user and the visited hotspot can negotiate a fresh session key with the shared secret *PMK*.

Fig. 4 shows the architecture of our mutual authentication. The architecture can be divided into three phases: association, credential verification, and the 4-way handshake. Note that the association phase and the 4-way handshake phase are provided by the Protected EAP. In the credential verification phase, the user and AS perform mutual authentication through a simple challenge-response method. After that, the user and the visited hotspot perform mutual authentication through the 4-way handshake phase.

VI. Security Analysis

A. Mutual Authentication and Key Agreement

Our scheme provides the mutual authentication between a user and a visited hotspot. Our mutual authentication is based on the proposed access credential management scheme and the EAP method. When the user connects to a visited hotspot, it first performs mutual authentication with AS. AS can verify the user with its access credential. The random number r is a challenge and encrypted by the symmetric key shared between the user and AS. If AS returns the correct challenge number, the user finishes authentication with AS. After the mutual authentication between the user and AS, both the user and the visited hotspot know the shared secret *PMK*, and then they negotiate a session key with *PMK*. In this way, the user and the visited hotspot finish the mutual authentication between them. Therefore, the mutual authentication and session key agreement is done safely in our scheme.

B. Anonymity and Unlinkability

Anonymity set is a metric to quantify the degree of unlinkability between access credentials and users' identities. Anonymity set is a notion similar to k-anonymity: a user has k-anonymity if its actions cannot be distinguished from the ones of other $k-1$ users; all k users form the anonymity set. In this paper, the anonymity set of an access credential is a user identity set which contains all the possible identities linking to the access credential.

After participating a credential exchange protocol, the anonymity set of the user's new access credential is an union of the anonymity sets of old access credentials owned by honest users who participate this credential exchange protocol. Thus, the anonymity set of an access credential is increased or unchanged after the access credential participates in a credential exchange protocol.

The unlinkability among access credentials owned by the same user is based on the unlinkability property of credential exchange protocol. The credential exchange protocol owns the unlinkability property when there are at least two honest participants in the protocol.

C. Mutual Accountability

Our scheme achieves user accountability without comprising users' anonymity. On the one hand, our proposed scheme can prevent users from sharing their access credentials because sharing an access credential means its access credential maybe belong to the other one forever. The one receiving the shared access credential can immediately participate in a credential exchange protocol to obtain a new anonymous access credential only for its own use. On the other hand, AS can detect selfish attacks and prevent misbehaving credential holders from accessing public hotspots. First, the number of valid access credentials owned by users are not larger than that of registration transactions created by the users because of anti-double-spending property of the blockchain and the verification path rule defined in our scheme. Second, AS can revoke the misbehaving access credentials through blacklist updating, but it cannot link these access credentials to users' identities before they leave the system. Third, the verification path rule makes accountability possible. The SGX-based mixer serving users will refuse to let blacklisted credential holders participate in the credential exchange protocol. Otherwise, other honest users' new credentials may be the

TABLE I
SIZE EVALUATION OF DIFFERENT TRANSACTIONS

	Deposit	Registration	Blacklist update	Credential revocation	Withdrawal	Participation	Credential exchange
Size (bytes)	368	307	267	235	242/276*	287	$214 \times (k+1) + 21$*

*The withdrawal transaction for an honest user is 242 bytes and that for a malicious user is 276 bytes. The size of the credential exchange transaction is proportional to the number of participants.

blacklisted ones after the credential exchange protocol. Finally, the deposit transaction increases the cost of misbehavior. When users withdraw their deposits, their identities are disclosed and misbehaving access credential holders will be identified.

AS cannot abuse accountability mechanism to deanonymize users without users' awareness. Since AS must update the state of the blacklist in the blockchain. In the case that the credential exchange transaction containing the blacklisted access credential is broadcasted in the Bitcoin network or included in the blockchain, the secure enclave in the SGX-based mixer will monitor the blacklist update transaction to make sure that the authentication server blacklists the access credential provided by the secure enclave. To sum up, AS must update blacklist in the blockchain when it performs user accountability.

VII. Efficiency Analysis and Performance Evaluation

In this section, we first analyze communication, storage, and computation overhead for our scheme. Then, we compare the authentication time of our scheme at the authentication server and the user side with FARB, which is the most efficient authentication scheme supporting blacklist without TTPs.

A. Communication Overhead

In our scheme, the main communication overhead for a user is incurred by managing access credentials through the blockchain. In the following, we first estimate the size of every transaction defined before. Then, we compute the communication overhead incurred by performing protocols.

A user, running a lightweight Bitcoin client, will download all block headers of the blockchain and transactions corresponding to access credentials. We will use the method in [27] to evaluate the space consumption of transactions in our scheme. First, there is a fixed 10 bytes in a transaction without any input and output. The input and output of every P2PKH transaction contribute about 180 bytes and about 34 bytes, respectively. The input and output of every 2-of-2 multisig transaction yield about 187 bytes and about 133 bytes, respectively. Every marker output results in 11-83 bytes. The marker output in registration transaction produces 83 bytes and that with NULL payload 11 bytes. In blacklist update transaction, the marker output occupies 43 bytes if its payload is a Merkle root. In the participation transaction, the marker output generates 63 bytes when considering that $pid||r$ is 32 bytes. Thus, we can evaluation the size of every transaction, as shown in Table I.

Moreover, we observe that the main communication overhead for a user in every protocol is incurred by the transactions to manage access credentials. We outline the communication overhead in each protocol in Table II. We observe that the main communication overhead of a user is incurred by downloading the block headers and performing credential exchange protocols.

TABLE II
COMMUNICATION OVERHEAD FOR A USER

Protocols	Comm. overhead (bytes)
Mutual authentication	253
Credential exchange	$214 \times (k+1) + 494$
User registration	1,334
Credential revocation (a malicious user)	761
Credential revocation (an honest user)	928
Block header synchronization	11,520 per day

The design of our scheme is based on two facts: (1) Basically, there are only two kinds of blockchain operations, i.e., write to or read from the blockchain. While the write latency is significant, the read latency is essentially zero. Because we can read the blockchain-related data from any full node (the node that holds the entire blockchain and synchronizes online), the permissionless blockchain is "read-friendly" and can support a large number of concurrent reads. (2) Different management and authentication tasks for Wi-Fi access control require different time scales. It is normal for a user to take tens of minutes to complete the registration in a new system, but the authentication for a Wi-Fi hotspot access should be finished in microsecond time scale.

Time-scale requirements for credential exchange tasks are a bit complicated. The frequency of credential exchange depends on how sensitive users are to their privacy. If users are not extremely sensitive to their privacy, they can change their anonymous credentials with a low frequency (for example, tens of minutes). In this case users can utilize on-chain credential exchange schemes. If users are extremely sensitive to their privacy, they may want to change their anonymous credentials with a very high frequency (for example, under a minute). In this case users can utilize off-chain credential exchange schemes described in Section V-A.

For mutual authentication tasks based on credentials, all of them only involve reads from the blockchain. Therefore, our system can support fast authentications for users' Wi-Fi access. These authentications can be done concurrently (by reading blockchain-related data from different full nodes), so there is no bottleneck in the authentication process.

B. Storage Overhead

In our scheme, the storage overhead for a user consists of two parts: the overhead incurred by storing public and private keys, and the overhead incurred by storing the information of the blockchain.

Every user needs to store AS's public key and some key pairs owned by itself. In our scheme, we use ECDSA with the

TABLE III
COMPLEXITY ANALYSIS FOR A USER AND AS

Protocols	Computations for a user	Computations for the authentication server
Mutual authentication	1S+1Enc+1Dec	1SV+1Enc+1Dec
Credential exchange	1S+1Enc	NULL
User registration	2S+1SV	1S+1SV
Blacklist update	NULL	1S
Credential revocation (a malicious user)	2S+1SV	1S+2SV
Credential revocation (an honest user)	3S+1SV	1S+2SV
Blockchain synchronization	block header verification	unconfirmed transaction verification + block verification

*In the table, S is the operation for ECDSA signature, SV is the operation for ECDSA signature verification, Enc is the operation for ECIES encryption, and Dec is the operation for ECIES decryption. NULL means that one party does not participate in the protocol.

secp256k1 curve. In ECDSA, a public key is 65 bytes, and the corresponding private key is 32 bytes. Thus, a pair of public and private key has 97 bytes. In our scheme, a user at least stores $1 + l$ pairs of public and private keys: one for withdrawing the deposit and the other ones for recent l access credentials. Given $l = 10$, the storage overhead incurred by storing AS's public key and key pairs has at least 1,132 bytes.

A user running a lightweight Bitcoin client needs to store the whole block headers and some $txid$s, such as $txid$ for the deposit transaction. Every block header is 80 bytes and there are 720,550 block headers as of January 2022. It is about 57.6 MB to store the block headers. Consider that a $txid$ is 32 bytes, it is clear that the main storage overhead for a user is incurred by storing the block headers.

C. Computation Overhead and Performance Evaluation

In our scheme, the operations for ECDSA and ECIES are the most expensive operations when each protocol is performed. Thus, we outline the major operations in each protocol in Table III. Notice that the time to query the access credential set is negligible for AS.

Similar to FARB, our mutual authentication protocol in this paper also has constant computational complexity, both on AS and user side. However, FARB cannot prevent users from sharing their access credentials.

We now compare the mutual authentication performance of our scheme with FARB. We benchmarked the time required for ECDSA and ECIES operations in our scheme and pairing operations in FARB. We obtained the benchmark for AS on a Dell Optiflex 7050 desktop computer with an Intel i7-7700 3.60 GHz CPU and 8 GB DRAM running Windows 10 as the operating system; we obtained the benchmark for mobile users on a Xiaomi 8 smartphone with a Qualcomm Snapdragon 845 processor and 6 GB RAM running Android 10 as the operating system. The test code for our scheme is based on Crypto++ library[5] and the test code for FARB is based on Pairing-Based Cryptography (PBC) library,[6] using type d224 pairing.

The experimental results are shown in Table IV. For FARB, it takes a user and AS about 85.45 ms and 106.13 ms, respectively. For our scheme, it only takes a user and AS about 5.65 ms and 3.96 ms, respectively. Thus, our proposed scheme is more efficient than the existing schemes with similar functionalities.

TABLE IV
BENCHMARKS AND AUTHENTICATION TIME AT THE USER AND AS SIDE

	User (ms)	Authentication server (ms)
ECDSA signature	1.29	0.56
ECDSA verification	4.79	2.07
ECIES encryption	2.50	1.09
ECIES decryption	1.86	0.80
E1	1.93	0.91
ET	0.009	0.004
EP	8.96	4.87
FARB	85.45	106.13
Proposed scheme	5.65	3.96

VIII. CONCLUSION

In this paper, we have presented a novel mutual authentication scheme by utilizing the Bitcoin blockchain and SGX-based mixing techniques to simultaneously achieve security, anonymity, accountability, and efficiency for public Wi-Fi hotspot access without involving any trusted third party. The security analysis demonstrates that our scheme can meet these requirements. Moreover, efficiency analysis and performance evaluation show that our mutual authentication scheme have low communication and computation overhead both on the user side and the authentication server side.

[5] https://www.cryptopp.com/

[6] https://crypto.stanford.edu/pbc/

REFERENCES

[1] Y. Niu, L. Wei, C. Zhang, J. Liu, and Y. Fang, "An anonymous and accountable authentication scheme for Wi-Fi hotspot access with the Bitcoin blockchain," in *Proc. IEEE/CIC Int. Conf. Commun. China*, 2017, pp. 1–6.

[2] C. Pisa, A. Caponi, T. Dargahi, G. Bianchi, and N. Blefari-Melazzi, "WI-FAB: Attribute-based WLAN access control, without pre-shared keys and backend infrastructures," in *Proc. 8th ACM Int. Workshop Hot Topics Planet-Scale Mobile Comput. Online Social Netw.*, 2016, pp. 31–36.

[3] B. Potter, "Wireless hotspots: Petri dish of wireless security," *Commun. ACM*, vol. 49, no. 6, pp. 50–56, Jun. 2006.

[4] P. P. Tsang, M. H. Au, A. Kapadia, and S. W. Smith, "Blacklistable anonymous credentials: Blocking misbehaving users without TTPs," in *Proc. 14th ACM Conf. Comput. Commun. Secur.*, 2007, pp. 72–81.

[5] J. Camenisch and A. Lysyanskaya, "An efficient system for non-transferable anonymous credentials with optional anonymity revocation," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 2001, pp. 93–118.

[6] P. Kyasanur and N. H. Vaidya, "Selfish MAC layer misbehavior in wireless networks," *IEEE Trans. Mobile Comput.*, vol. 4, no. 5, pp. 502–516, Sep./Oct. 2005.

[7] N. Funabiki, T. Nakanishi, H. Takahashi, K. Miki, and J. Kawashima, "A proposal of anonymous IEEE802.1X authentication protocol for wireless networks," in *Proc. 2nd IEEE Workshop Secure Netw. Protoc.*, 2006, pp. 26–31.

[8] W. Lou and K. Ren, "Security, privacy, and accountability in wireless access networks," *IEEE Wireless Commun.*, vol. 16, no. 4, pp. 80–87, Aug. 2009.

[9] D. He, S. Chan, and M. Guizani, "An accountable, privacy-preserving, and efficient authentication framework for wireless access networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1605–1614, Mar. 2016.

[10] L. Xi and D. Feng, "FARB: Fast anonymous reputation-based blacklisting without TTPs," in *Proc. 13th Workshop Privacy Electron. Soc.*, 2014, pp. 139–148.

[11] T. Nakanishi and T. Kanatani, "Efficient blacklistable anonymous credential system with reputation using a pairing-based accumulator," *IET Inf. Secur.*, vol. 14, no. 6, pp. 613–624, Nov. 2020.

[12] A.-E. Panait and R. F. Olimid, "On using zk-SNARKs and zk-STARKs in blockchain-based identity management," in *Proc. Int. Conf. Inf. Technol. Commun. Secur.*, 2020, pp. 130–145.

[13] H. Halpin, "Nym credentials: Privacy-preserving decentralized identity with blockchains," in *Proc. Crypto Valley Conf. Blockchain Technol.*, 2020, pp. 56–67.

[14] E. Samir, H. Wu, M. Azab, C. Xin, and Q. Zhang, "DT-SSIM: A decentralized trustworthy self-sovereign identity management framework," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 7972–7988, Jun. 2022.

[15] P. Dunphy and F. A. Petitcolas, "A first look at identity management schemes on the blockchain," *IEEE Secur. Privacy*, vol. 16, no. 4, pp. 20–29, Jul./Aug. 2018.

[16] R. N. Zaeem et al., "Blockchain-based self-sovereign identity: Survey, requirements, use-cases, and comparative study," Center for Identity, Univ. of Texas at Austin, Austin, Texas, USA, Tech. Rep. UTCID Report #21-06, 2021.

[17] Y. Yao, X. Chang, J. Mišić, V. B. Mišić, and L. Li, "BLA: Blockchain-assisted lightweight anonymous authentication for distributed vehicular fog services," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3775–3784, Apr. 2019.

[18] Z. Lu, Q. Wang, G. Qu, H. Zhang, and Z. Liu, "A blockchain-based privacy-preserving authentication scheme for VANETs," *IEEE Trans. Very Large Scale Integration Syst.*, vol. 27, no. 12, pp. 2792–2801, Dec. 2019.

[19] X. Li, J. Liu, M. S. Obaidat, P. Vijayakumar, Q. Jiang, and R. Amin, "An unlinkable authenticated key agreement with collusion resistant for VANETs," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 7992–8006, Aug. 2021.

[20] X. Liu, A. Yang, C. Huang, Y. Li, T. Li, and M. Li, "Decentralized anonymous authentication with fair billing for space-ground integrated networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 7764–7777, Aug. 2021.

[21] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton, NJ, USA:Princeton Univ. Press, 2016.

[22] M. Belotti, N. Božić, G. Pujolle, and S. Secci, "A vademecum on blockchain technologies: When, which, and how," *IEEE Commun. Surv. Tut.*, vol. 21, no. 4, pp. 3796–3838, Oct.–Dec. 2019.

[23] J. Huang, D. He, M. S. Obaidat, P. Vijayakumar, M. Luo, and K.-K. R. Choo, "The application of the blockchain technology in voting systems: A review," *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–28, Apr. 2022.

[24] M. Rosenfeld, "Overview of colored coins," White Paper, 2012. [Online]. Available: https://bitcoil.co.il/BitcoinX.pdf

[25] T. Ruffing, P. Moreno-Sanchez, and A. Kate, "Coinshuffle: Practical decentralized coin mixing for bitcoin," in *Proc. Eur. Symp. Res. Comput. Secur.*, Wroclaw, Poland, 2014, pp. 345–364.

[26] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*. Berlin, Germany: Springer, 2006.

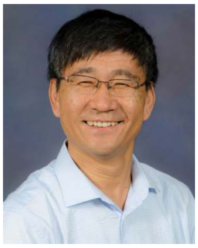
[27] M. Bartoletti and L. Pompianu, "An analysis of Bitcoin OP_RETURN metadata," in *Proc. Int. Conf. Financial Cryptography Data Secur.*, 2017, pp. 218–230.

Yukun Niu received the B.E. degree in information security and the Ph.D. degree in information and communication engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2012 and 2022, respectively. He is currently a Postdoctoral Fellow with the Endogenous Security Research Center, Purple Mountain Laboratories (PML), Nanjing, China. His research interests include applied cryptography, blockchain, privacy, and consensus in endogenous safety.

Lingbo Wei (Member, IEEE) received the B.S. degree in mathematics from Shaanxi Normal University, Xi'an, China, in 2001, the M.S. degree in cryptography from Xidian University, Xi'an, China, in 2005, and the Ph.D. degree in information security from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2009. From 2009 to 2014, she was a Postdoctoral Fellow with Beihang University, Beijing, China, and Shanghai Jiao Tong University, Shanghai, China. From 2014 to 2020, she was an Associate Professor with the School of Information Science and Technology, University of Science and Technology of China, Hefei, China. She is currently a Research Fellow with Hefei Comprehensive National Science Center, China. Her research interests include blockchain and applied cryptography.

Chi Zhang (Member, IEEE) received the B.E. and M.E. degrees in electrical and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1999 and 2002, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Florida, Gainesville, FL, USA, in 2011. In 2011, he joined the School of Information Science and Technology, University of Science and Technology of China, Hefei, China, as an Associate Professor. His research interests include the areas of network protocol design and performance analysis and network security particularly for wireless networks and blockchains. He was the recipient of the 7th IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award.

Jianqing Liu (Member, IEEE) received the B.Eng. degree from University of Electronic Science and Technology of China, Chengdu, China, in 2013, and the Ph.D. degree from The University of Florida, Gainesville, FL, USA, in 2018. He is currently an Assistant Professor with the Department of Computer Science, NC State University, Raleigh, NC, USA. His research interests include wireless communications and networking, security, and privacy. He was the recipient of the U.S. National Science Foundation Career Award in 2021. He was also the recipient of several best paper awards, including the 2018 Best Journal Paper Award from IEEE Technical Committee on Green Communications & Computing (TCGCC).



Yuguang Fang (Fellow, IEEE) received the M.S. degree from Qufu Normal University, Jining, China, in 1987, the Ph.D. degree from Case Western Reserve University, Cleveland, OH, USA, in 1994, and the second Ph.D. degree from Boston University, Boston, MA, USA, in 1997. Since August 2022, he has been the Chair Professor of Internet of Things with the Department of Computer Science, City University of Hong Kong, Hong Kong. In 2000, he joined the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA, and has been a Full Professor since 2005. He held the University of Florida Research Foundation Professorship from 2006 to 2009, a Changjiang Scholar Chair Professorship with Xidian University, Xi'an, China, from 2008 to 2011, and has been with Dalian Maritime University, Dalian, China, since 2015. From 2009 to 2012, he was the Guest Chair Professorship with Tsinghua University, Beijing, China. He was the recipient of the U.S. National Science Foundation Career Award in 2001, Office of Naval Research Young Investigator Award in 2002, 2015 IEEE Communications Society CISTC Technical Recognition Award, 2014 IEEE Communications Society WTC Recognition Award, Best Paper Award from IEEE ICNP in 2006, 2010–2011 UF Doctoral Dissertation Advisor/Mentoring Award, 2011 Florida Blue Key/UF Homecoming Distinguished Faculty Award, and 2009 UF College of Engineering Faculty Mentoring Award. He is the Editor-in-Chief of the IEEE Transactions on Vehicular Technology, was the Editor-in-Chief of the IEEE Wireless Communicationsfrom 2009 to 2012, and was on several Editorial Boards of journals, including the IEEE Transactions on Mobile Computing from 2003 to 2008 and since 2011, the IEEE Transactions on Communications from 2000 to 2011, and the IEEE Transactions on Wireless Communications from 2002 to 2009. He has actively participated in conference organizations, such as serving as the Technical Program Co-Chair for IEEE INOFOCOM'2014 and Technical Program Vice-Chair for IEEE INFOCOM'2005. He is a Fellow of the American Association for the Advancement of Science.