Hindawi
Security and Communication Networks
Volume 2021, Article ID 3025774, 17 pages
https://doi.org/10.1155/2021/3025774

Research Article

Covert Communication Scheme Based on Bitcoin Transaction Mechanism

Shanyun Huang[1,2], **Wenyin Zhang**[2], **Xiaomei Yu**[1], **Jiuru Wang**[2], **Wanshui Song**[1,2] and **Bei Li**[2,3]

[1]*School of Information Science and Engineering, Shandong Normal University, Jinan 250000, China*
[2]*School of Information Science and Engineering, Linyi University, Linyi 276000, China*
[3]*School of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266000, China*

Correspondence should be addressed to Wenyin Zhang; zhangwenyin@lyu.edu.cn and Xiaomei Yu; 111110@sdnu.edu.cn

Received 14 September 2021; Revised 10 November 2021; Accepted 15 November 2021; Published 10 December 2021

Academic Editor: Marimuthu Karuppiah

Copyright © 2021 Shanyun Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the unique characteristics of blockchain, such as decentralization, anonymity, high credibility, and nontampering, blockchain technologies have become an integral part of public data platforms and public infrastructure. The communication between the stakeholders of a given blockchain can be used as a carrier for covert communication under cover of legal transactions, which has become a promising research direction of blockchain technology. Due to the special mechanism of blockchain, some traditional blockchain covert communication schemes are not mature enough. They suffer from various drawbacks, such as weak concealment of secret information, cumbersome identification and screening of special transactions, poor availability, and low comprehensive performance. Therefore, this paper designs a scheme of covert communication in the Bitcoin blockchain, which takes normal transactions as a mask and leverages the Bitcoin transaction mechanism to embed secret information in the public key hash field. Specifically, we propose a novel key update mechanism combined with the hash algorithm to construct a covert channel. It ensures security and can update the channel to prevent the related problems caused by address reuse. We are taking advantage of the feature of Bitcoin that cannot be double-spent to solve the problem of burning bitcoin when paying bitcoin to a fake public key hash. In our scheme, both parties to the communication are anonymous, and the attacker cannot detect the covert data or track the transaction and address. Our proposed scheme was tested in real Bitcoin blockchain network, and the experimental results were analyzed to verify its security, availability, and efficiency.

1. Introduction

Since Lampson [1] first put forward the concept of covert communication in 1973, covert communication technology has made considerable progress. Information hiding is an important method to realize covert communication. The Proceedings of the First International Workshop on Information Hiding [2] was held in Cambridge University from May 30 to June 1, 1996, marking the formal emergence and rapid development of information hiding. Malaichamy et al. [3] proposed a novel encryption of text messages using a twofold approach to transmit text safely. Deverajan et al. [4] present a public key encryption with equality test based on DLP with double decomposition problems over near-ring, which can transmit and share data safely and flexibly in the Internet of Things environment. Because of the vulnerability of wireless sensor networks, Palanisamy et al. [5] present communication trust and energy aware (CTEA) routing protocol that make use of the proposed trust model to mitigate the effects of badmouth and energy drain attacks. The concept of steganography in information hiding is to hide the secret information in some carriers to protect the secret information. One of the modern steganography methods is network steganography. Other methods include digital media steganography, physical steganography, and encryption system steganography, among others.

Network steganography is based on constructing a secure covert channel, an important branch of information hiding. The construction of the covert channel is the basis of covert communication. In 1996, Handel and Sandford [6] introduced the covert channel into computer networks for the first time, marking the rapid development of covert communication based on traditional network TCP/IP architecture. In covert communication, the sender and receiver need to establish a covert channel in the normal communication channel for covert communication while ensuring the integrity of the communication information and not being noticed by anyone other than the communicating parties. The two main types of covert network channels are covert storage channels (CSCs) based on the space domain and covert time channels (CTCs) based on the time domain [7]. However, with the development of covert communication, many detection methods have also been developed [8, 9], making covert communication under the traditional network face more risks. The covert storage channels have the disadvantage of insufficient concealment. The covert time channels have a small capacity and are relatively restricted by the network. In addition, some authentication protocols have been proved to be no longer safe and practical [10].

Due to practical problems such as weak concealment, low transmission efficiency, and large limitations of covert communication under traditional networks, researchers have turned their attention to the blockchain network. They are committed to building a more concealed, higher transmission efficiency, difficult detection, and more robust covert channel for covert communication. Blockchain is derived from the underlying technology of Bitcoin. In 2008, a scholar named "Satoshi Nakamoto" proposed a digital currency called Bitcoin [11]. In the absence of any authoritative intermediary organization to coordinate, people who do not trust each other can directly use bitcoin to pay. Blockchain is an emerging technology covering many fields, such as distributed systems and the Internet of things [12]. In addition, cloud data and log file systems [13] can also be combined with blockchain to achieve better results. Some scholars have sorted out some articles on blockchain solutions in recent years through complex networks [14], provided visual graphics, and analyzed the future development direction of blockchain [15]. Blockchain has the characteristics of nontampering, anonymity, decentralization, etc. It has its unique advantages in various fields, and these characteristics combined with the structural characteristics of the blockchain itself are more conducive to the construction of covert channels through information hiding to realize the covert transmission of information. The information published on the blockchain cannot be censored, the security and credibility are high, the amount of data is huge, and both parties to the transaction are anonymous. Therefore, hiding information on the blockchain and then conducting covert communication has inherent advantages. In the future, as a public platform and public infrastructure, the blockchain will easily be used as a carrier of covert communication, through the characteristics of the blockchain or under cover of transactions for covert communication. However, the combination of blockchain and covert communication has achieved very little so far, mainly limited to the theoretical level. Most of the existing blockchain covert communication schemes are not perfect and mature.

Therefore, we design a new covert communication scheme based on the Bitcoin transaction mechanism, which uses Bitcoin public key hash to hide secret information. The scheme uses keys to ensure the security of secret information, establishes a covert channel between the receiver and the sender, uses the hash algorithm to realize the dynamic update of the address while realizing the identification of special transactions, which ensures the security of the covert channel and avoids a series of risk problems caused by address reuse. Our main contributions are listed as follows:

(1) We have established a general model of covert communication on the blockchain and summarized and analyzed the three important links of covert communication: secret information embedding, transaction identification and screening, and secret information extraction.

(2) We improved the information embedding method based on the Pay-To-Fake-Public-Key-Hash (P2FPKH) in the Bitcoin blockchain, ensured the concealment of secret information through encryption algorithms, and formulated the rules for embedding information.

(3) The iterative hash method is used to update the key. The communication address is updated simultaneously to ensure that the covert communication between unrelated addresses is securely carried out under the condition that the key is not leaked and address reuse is avoided.

(4) It is taking advantage of the feature of Bitcoin that cannot be double-spent to avoid generating UTXO that can never be spent by the fake public key hash and will not add permanent and unnecessary burdens to Bitcoin miners.

(5) We designed a complete covert communication scheme based on the bitcoin transaction mechanism through induction and the proposed method, conducted experiments on the real Bitcoin network, and analyzed the experimental results.

2. Related Works

For the problem of covert communication in the blockchain, some domestic and foreign researchers have made attempts in recent years. In 2012, Warren [16] first proposed the concept of communication through a blockchain network and proposed a messaging system based on the Bitcoin protocol, which simulates the Bitcoin network for message transmission and guarantees the message authenticity and completeness. Nevertheless, this system is an ideal communication application, which is separated from the blockchain network of Bitcoin. In 2015, Sleiman et al. [17] proposed a system that uses arithmetic coding to hide secret

information in the value field of Bitcoin, which can carry out covert communication, but its transmission efficiency is low, and the coding method is not safe for hiding information. Moreover, the value field is special after being coded, and it is easy to detect and track. In 2018, Partala [18] proposed a system called "BLOCCE," which is the first in the true sense to use the blockchain for covert communication. It uses the least significant bit (LSB) of the transaction address to establish an ideal covert transmission model in the Bitcoin network so that both parties can carry out covert communication securely through the model. Its transmission security is relatively high, but the transmission efficiency is relatively low.

In 2018, Frkat et al. [19] designed a scheme to transmit information in a blockchain network to drive a botnet, hiding instructions in the signatures of bitcoin transactions, called "chain channels." It enables the controlled host to search for instructions in the Bitcoin network automatically. The bot program completes related operations through these instructions. This scheme requires a relatively high degree of concealment of the bot program, and the key update mechanism is too cumbersome. In 2019, Li et al. [20] proposed a blockchain network covert channel model, using formal methods to model and prove the anti-interference and anti-tampering characteristics of the blockchain, and constructed a covert channel of blockchain network based on the transaction time interval, and laid a theoretical foundation for the practical application of a new network covert channel based on the blockchain environment. In 2020, Gao et al. [21] proposed a covert communication scheme based on the kleptography algorithm. Through the kleptography algorithm, the sender sets two special signatures in the blockchain transaction and encrypts the secret information with the public key. After the receiver detects the special transaction, the sender's private key can be extracted through the signature for decryption. This scheme can achieve better results in concealed communication. However, due to the limitation of its signature, both parties must communicate on an inherent address, which increases the possibility of being monitored and analyzed, and re-exchanging keys outside the chain increases the risk of channel exposure. Zhang et al. [22] improved "BLOCCE." The improved system is called "V-BLOCCE." It uses base58 code to encode secret information and uses each base58 encoded information as the least significant bit of the address, which slightly increases the information transmission capacity in each transaction relative to "BLOCCE." This scheme uses the OP_RETURN field to store the address sequence and index information, but the easy analysis and nongeneral nature of the OP_RETURN field result in the low availability of the scheme. Song and Peng [23] improved "BLOCCE." The improved system is called "BLOCCE+." The scheme changes the least significant bit of the address where the information is stored to the least α bits, which slightly increases the embedding capacity of each transaction. However, the calculation of the least α bit brings much computational consumption. Zhang et al. [24] used the intelligent contract of voting and bidding to transmit secret information, used an encryption algorithm and two rounds of protocols to ensure data privacy, and designed corresponding information embedding and transmission methods for different scenarios, which is a new covert communication scheme. She et al. [25] put forward a double steganography model which combines blockchain and Interplanetary File System (IPFS). The model consists of image steganography and plain text steganography. The model has high anti-detection performance and ensures the concealment of communication. Guo et al. [26] proposed a useable and secure covert channel on Monero. The complete anonymity of Monero effectively protects the relationship between the sender and the receiver and can safely conduct covert communication.

In covert communication, the method for the receiver to identify a special transaction is generally a transaction identification mechanism based on a fixed address or a transaction identification mechanism based on a data embedding algorithm, in which security risks are significant, and the calculations are complicated. In order to enable the receiver to accurately, safely, and quickly locate the special transaction where the sender hides secret information, some scholars have done relevant research. Tian et al. [27] proposed a new blockchain covert channel construction scheme-dynamic label chain, called "DLchain," and it adopts a dynamic label mechanism to identify and screen transactions. In order to ensure the concealment of dynamic labels, a dynamic label generation algorithm based on the statistical distribution of real transaction data is designed. However, transactions must also be identified from vast amounts of data. While the security level is improved, the efficiency of recognition is not improved. Si et al. [28] proposed a transaction identification algorithm based on the dynamic label. The scheme uses the HMAC algorithm to construct a special address that carries label information so that the receiver can quickly identify special transactions that carry hidden information from the massive data of the blockchain ledger and realize the covert transmission of data in an open blockchain environment. Nevertheless, it does not fundamentally improve the efficiency of identification. Although the recognition process is optimized, it still has to undertake cumbersome screening tasks.

3. Preliminaries

3.1. Standard Script Type of Bitcoin. There are currently five standard script types that are used and accepted on the Bitcoin network for transactions. The standard script types include Pay-to-Public-Key (P2PK), Pay-to-Public-Key-Hash (P2PKH), Multi-Signature, Pay-to-Script-Hash (P2SH), and OP_RETURN [29]. Each different script type has a different transaction method on the Bitcoin blockchain. In addition to the above five types, in order to achieve segregated witness, the Bitcoin community has increased the transaction capacity of each block of Bitcoin by adding a new transaction script type bech32.

Bitcoin addresses are divided into three types. The first is the P2PKH format. This address starts with the number "1" and is the most common address format in Bitcoin, such as "1xxxxxx...." The second is the P2SH format, the address

starts with "3," such as: "3xxxxxx...." The last is the Bech32 format, which starts with "bc1," such as "bc1xxxxxx...." This format is an address format specially developed for Segregated Witness. Through a macro analysis of the usage of each address, the P2PKH format addresses have a large number of users and a large number of transactions. In order to hide information safely, the address in P2PKH format with the most users is selected to construct a covert channel. The process of generating this address is shown in Figure 1.

To generate a bitcoin address in P2PKH format, first, randomly generate the 256-bit binary number and use it as a private key. Then, use the private key to generate the public key through the SECP256k1 algorithm, and use the public key to generate the public key hash through the SHA256 algorithm and the RIPEMD160 algorithm. 0×00 is the version number of the P2PKH format address. Perform the double SHA256 algorithm on the public key hash and extract the first four bytes as the check code. Finally, the combined version number, public key hash, and check code are encoded by Base58 to generate a new bitcoin address.

It can be seen from the description of the bitcoin address generation process that the information can be stored in the custom public key hash field by abandoning the private key and customizing the public key hash. However, because this type of address does not have a private key, no one can obtain the dominance of this address, and any funds transferred into the address cannot be transferred out. As early as 2013, someone used a fake public key hash field to embed information on the Bitcoin network. The content of the information features in Mandela's biography [30]. Although the information is embedded in the blockchain, the effect of information hiding and the purpose of covert communication are not achieved. Because the information is not processed in any way, the information is directly stored in the public key hash field in hexadecimal form. Bitcoin network monitors and on-chain data analysts can easily obtain this information through hexadecimal transcoding.

3.2. Application of Encryption Algorithm and Hash Algorithm. AES is an advanced encryption standard in cryptography, also known as Rijndael encryption method, a block encryption standard adopted by the US federal government. AES is a commonly used symmetric encryption algorithm with high security. In order to adapt to the capacity of carriers in the blockchain, the key length of the AES encryption algorithm we chose is 128 bits, and the ciphertext generated is 128 bits. Supposing the secret information is m, and the AES symmetric encryption scheme is SE_{AES}, SEAES = {Gen, Enc, Dec}. *Gen* is the key generation algorithm, and the key k with the length of 128 bits can be generated by setting the security parameter λ, which is expressed as $k \longleftarrow \mathrm{Gen}(\lambda)$. *Enc* is the encryption algorithm, input the key k and secret information m, and output ciphertext c, expressed as $c \longleftarrow \mathrm{Enc}(m, k)$. *Dec* is the decryption algorithm. Input the key k and ciphertext c, and output the secret information m in plaintext, expressed as $m \longleftarrow \mathrm{Dec}(c, k)$.

Since symmetric encryption is different from asymmetric encryption, if the same key is used multiple times, the security will be reduced in theory. In the actual application strategy, the key needs to be changed regularly or one encryption at a time. In order to implement the key update safely, we introduce a hash algorithm, use the hash function to achieve a secure key update, and guarantee one key at a time. Commonly used hash algorithms include MD5, SHA1, SHA256, Keccak256, and RIPEMD160. The anti-collision mechanism of MD5 and SHA1 has been broken and is considered unsafe. In addition, the length of the output generated by hash should be 256 bits because the length of the Bitcoin private key is 256 bits. The output length of Keccak256 and SHA256 is 256 bits. SHA256 algorithm is widely used in Bitcoin system and has high security. So, we chose the SHA256 hash algorithm. The definition of the hash function is: mapping an input of arbitrary length to an output of a fixed length, that is, $\{0,1\}^n \longleftarrow \{0,1\}^*$. The mapping process of the hash function is irreversible. Suppose the hash function is SHA256, the function's input is s, and the output is h, expressed as $h \longleftarrow \mathrm{SHA256}(s)$. It is easy to calculate h if s is known, but it is challenging to calculate s if h is known. The ideal hash function is anti-collision. Namely, the mapping from s to h is unique. In addition, the output of the hash function can be regarded as random without knowing the input, which is a fundamental property. Set a security parameter λ and use the key generation algorithm to generate the key k. After completing a communication, the output obtained by hashing the key k is used as the new key k, and the iterative hashing is performed in this way to realize the key update. An obvious disadvantage of this method is that the initial key k determines the security, and the security of the key k must be guaranteed. However, the sequence obtained by iteratively hashing k can continuously and uniquely identify a 256-bit string, and this feature can be cleverly applied to our proposed scheme. Algorithm 1 shows the general process of iterative hashing.

Only the idea of iterative hashing is given here, and this method will be improved according to the requirements of the actual scheme.

3.3. UTXO That Can Never Be Spent. The UTXO model is adopted in the Bitcoin network. UTXO refers to unspent transaction outputs. If an address contains several bitcoins, but the address's private key is missing or does not exist, the address cannot be used as the input to output bitcoins, and a UTXO that can never be spent will be generated. The Bitcoin community believes that this UTXO that can never be spent will always remain in the UTXO pool maintained by the miners, which will slow down the operation of the Bitcoin infrastructure. The carrier of our covert communication scheme is the public key hash, which uses Pay-To-Fake-Public-Key-Hash to fake a public key hash to generate the corresponding special address. There is no private key for this special address and when this special address receives bitcoins, it will generate a UTXO that can never be spent. In the case of covert communication, such a UTXO that can

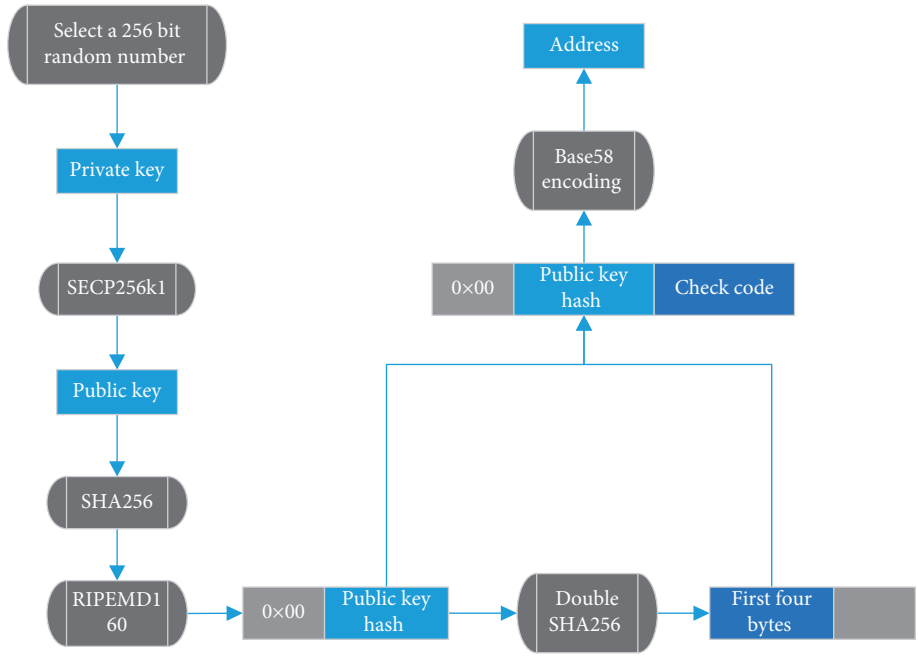

Figure 1: Generation process of bitcoin address of P2PKH format.

Input: Security parameter λ.
Output: Key k.
(1) $k \longleftarrow \text{Gen}(\lambda)$;
(2) After using k;
(3) $k \longleftarrow \text{SHA256}(k)$;
(4) After using k;
(5) $k \longleftarrow \text{SHA256}(k)$;
(6) .....;
(7) End;

Algorithm 1: Iterative hashing.

never be spent will threaten communication. The UTXO that has not been spent for a long time can be identified as the following four possible situations:

(1) The address owner's private key is lost, and the transfer transaction cannot be performed.

(2) The address balance is too small to support the handling fee required for transfer, or the address is abandoned and no longer used.

(3) The address is fake. There is no private key, and bitcoin cannot be consumed.

(4) The address owner has no intention to transfer bitcoins.

If the address has only participated in the transaction as the output, but has not participated in the transaction as the input, it belongs to the above four situations and cannot be distinguished into which one. As time passes, on-chain data analysts can find a set of addresses that meet the requirements by analyzing the data, which increases the risk of exposure to special addresses.

3.4. Bitcoin That Cannot Be Double-Spent. Double-spend means that an amount of money is reused twice. In a centralized payment system, the problem of double-spend is mainly solved by third-party institutions. In the decentralized blockchain, Bitcoin introduces the timestamp and UTXO model, which perfectly solves the problem of double-spend. In the Bitcoin system, after a transaction is a broadcast, it will first enter the UTXO pool maintained by the miner node and wait for the miner to verify it. After the verification is passed, the transaction enters the unconfirmed transaction pool and waits for packaging. The complete transaction information in the unconfirmed state can be obtained by querying the transaction address or transaction ID. In the Bitcoin network, double-spend can be created, but double-spend on the chain cannot occur. There are three addresses a, b, and c. The balance of address a is 1.1 bitcoins (consider transaction fee). If a is transferred to b 1 bitcoins, the transaction will be broadcast after the transaction T_{ab} is constructed, and the transaction T_{ab} will pass through the UTXO pool after verification, enter the unconfirmed transaction pool. Before T_{ab} has not been confirmed to be packaged on the chain, construct a transaction T_{ac} that transfers 1 bitcoin from address a to address c. After the transaction is broadcast, it enters the UTXO pool, and the verification conflicts cause the problem of double-spend. Bitcoin provides a solution to the problem of double-spend:

(1) When one of the transactions is packaged on the chain, the other transaction cannot be verified and will be regarded as an invalid transaction and discarded.

(2) If the two transactions are not packaged on the blockchain, the transaction with the higher fee will be packaged for Bitcoin miners to maximize their

benefits. This method solves the problem of double-spend and protects the rights and interests of Bitcoin miners. If the fee of T_{ab} is set to 1 sat/byte and the fee of T_{ac} is 2 sat/byte, then the transaction T_{ab} will be cleared, and T_{ac} will remain in the unconfirmed transaction pool waiting to be packaged.

4. Covert Communication on the Blockchain

This section mainly introduces the general model of covert communication on the blockchain and analyzes the key problem of covert communication.

4.1. The General Model of Covert Communication on the Blockchain. Figure 2 is the constructed covert communication on the blockchain model, which describes the general process of using the blockchain network for covert communication. The model includes two adversary attack models: off-chain traffic monitors and on-chain data analysts. The following describes the specific process of each step-in turn:

① The sender and receiver negotiate the key, algorithm, and other information required for communication before communication.

② The sender embeds the secret information m into the transaction by some method relying on the characteristics of the blockchain or under cover of the transaction to construct a special transaction with a normal transaction format.

③ Broadcast the special transaction until it is packaged into the block.

④ The receiver obtains all transaction records of all blocks from the blockchain.

⑤ The receiver obtains a special transaction through transaction identification or transaction screening.

⑥ The receiver views the special transaction and extracts the secret information.

The covert channel in the traditional network is centralized. The adversary can easily identify that the sender and the receiver are communicating by analyzing the network traffic, and it is difficult to hide the communication behavior. Exposure to communication behaviors will make it easier for adversaries to analyze the identities of both parties in communication. It is difficult for covert communication in the traditional network to hide communication behavior and achieve communication anonymity.

Unlike covert communication in the traditional network, covert communication on the blockchain does not require the communicating parties to establish a connected channel. Using the blockchain network to communicate, the sender passes the secret information through the characteristics of the blockchain and constructs a special transaction containing secret information under the cover of a legal transaction. The output address of the transaction does not need to be the address of the receiver of the secret information, it can be any address. The sender has to publish the special transaction containing the secret information to the blockchain network, and the receiver can find it through transaction identification and screening mechanism, and extract the secret information. The receiver and the sender do not have substantial detectable communication behavior for covert communication on the blockchain. The receiver can be any user who browses the blockchain data, completely concealing the receiver's existence, which hides the communication behavior. Suppose the sender's behavior of hiding secret information in the transaction is discovered. In that case, it cannot be defined as a communication behavior, or it may be a behavior of storing data information because the receiver cannot be found.

As we all know, the Bitcoin blockchain itself has anonymity, and it is difficult for an adversary to determine the true identity of both parties through a particular transaction. Some researchers have tried to analyze the entities in the bitcoin network by analyzing bitcoin transaction data and analyzing the true identity of the entities, but this has certain limitations. If the transaction data related to the entity is relatively small or the transaction characteristics are not prominent, it will be tough to analyze the entity and then analyze the true identity. Literature [31] introduces some methods and applications for analyzing Bitcoin entities. Regular coin shuffle or frequent exchange of transaction addresses will better ensure anonymity. Even if the sender constructs a special transaction containing secret information, the adversary will not be able to snoop on the sender's true identity. Covert communication on the blockchain network has obvious advantages over covert communication in the traditional network in terms of hiding communication behavior and anonymity.

4.2. Analysis of Key Problem. Covert communication on the blockchain is usually carried out on the public blockchain. Because the access authority of the public blockchain is low, both parties can establish a covert communication relationship. The public blockchain has a large amount of data, many users, and transactions, which is more conducive to the concealment of communication parties and special transactions. Through the establishment of the covert communication model on the blockchain, three key problems are summarized: how to embed secret information in transactions; how to quickly and efficiently identify or screen out special transactions that contain secret information; and how to safely extract secret information. Because the data on the public blockchain is public, the three links of information embedding, special transaction identification or screening, and information extraction must be carried out under safe conditions to ensure that on-chain data analysts cannot perceive the exposed transaction content.

(1) Information embedding: The existing blockchain applications with the broadest audience mainly include Bitcoin, Ethereum. Different covert communication schemes need to be designed in different blockchain networks. Embedding secret information in blockchain needs to consider many indicators such as concealment, capacity, and transmission

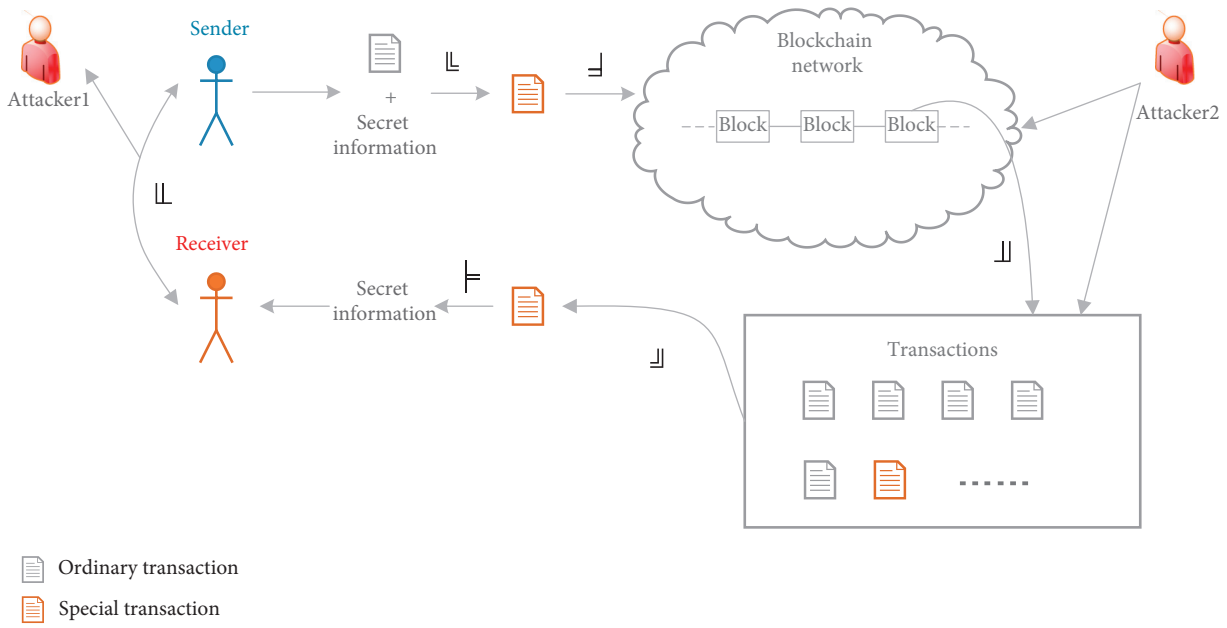

Figure 2: General model of blockchain covert communication.

efficiency. These indicators combine the characteristics of the blockchain to embed secret information to ensure the availability and practicability of the scheme. Embedding secret information into transactions needs to rely on the features of normal transactions and embedding secret information without changing the content and format of normal transactions, so that special transactions containing secret information and normal transactions are indistinguishable within the effective time of the information or within the polynomial time. Literature [29] summarizes 8 methods of embedding data in the Bitcoin blockchain. The capacity of embedded data ranges from 20 bytes to 1630 bytes, but its concealment and security are uneven, and it is difficult to escape the blockchain monitors and blockchain data analysts. Further improvements and designs are needed if we want to use the scheme mentioned in the literature to transmit secret information. We have improved the method of Pay-To-Fake-Public-Key-Hash mentioned in the literature and designed a complete scheme with high usability.

(2) Identification or screening of special transactions: The amount of information on the blockchain is enormous. For example, Bitcoin generates about 144 blocks with a data volume of about 144 MB every day, and each block contains many different transactions. After the sender embeds the secret information, the receiver needs to accurately, quickly, and efficiently identify or screen out special transactions containing secret information and ensure that no one other than the receiver can determine the special transaction. The most primitive way to identify transactions is to use fixed addresses. The communication parties exchange their respective addresses and construct special transactions through these addresses for covert communication. Although the screening efficiency is very high, address reuse undoubtedly increases the risk of covert channel exposure. There is also a method of screening special transactions based on the data embedding algorithm, which decrypts each carrier that may contain confidential information in all transactions in turn. This method has high concealment but extremely low screening efficiency. The last one is the method of the special transaction screening based on dynamic tags. Its essence is the same as the method of screening special transactions based on the data embedding algorithm. The difference is that it does not rely on secret information carriers to screen special transactions, and it has lower screening efficiency. We have designed a method of special transaction identification relying on the key and hash algorithm, which has high identification efficiency and strong concealment.

(3) Information extraction: The extraction of secret information is based on the embedding mechanism of secret information. The extraction of secret information depends on the off-chain information negotiation between the communicating parties before communication. The negotiated information includes an encoding algorithm or encryption algorithm and key and related information related to the entire communication scheme. The off-chain information negotiation must be carried out under secure conditions, and it is necessary to ensure one time or minimize off-chain interactions to ensure the safe extraction of secret information. Multiple off-chain information negotiations will increase the risk of key exposure.

5. Our Proposed Scheme

This section mainly introduces the symbols involved in the scheme and elaborates the specific process of the whole scheme from three aspects: embed secret information, extract secret information, key and address update. In addition, it is taking advantage of the feature of Bitcoin that cannot be a double-spend mechanism to withdraw the transaction, so that the method of Pay-To-Fake-Public-Key-Hash will not generate UTXO that can never be spent. Figure 3 is a model of our proposed scheme.

5.1. Notations. Table 1 shows some notations and their descriptions used in our scheme.

5.2. Embed Secret Information. In order to show the scheme more intuitively, imagine a scenario of covert communication between Alice and Bob on the Bitcoin blockchain. Alice is the sender, Bob is the receiver, and the secret information transmitted by the communication is m. Alice and Bob first negotiate the key K and symmetric key k and the used algorithm through the off-chain channel for covert communication, as shown in Figure 4. The scheme requires a relatively high degree of security for the key K, and we set its length as 256 bits. It is worth noting that the offchain information negotiation is only conducted once, ensuring the security of covert communication.

The length of the symmetric key k is 128 bits, Alice uses the key k to encrypt the secret information m to obtain the ciphertext c, and the length of the ciphertext c is 128 bits. The public key hash length of the generated bitcoin address is 160 bits, and the ciphertext c is combined with the 32-bit random number r as the special public key hash $Hash\_Pk_s$. The special address A_s is obtained by encoding the special public key hash *Hash Pks* through Base58Check. Use the key K as the private key to generate an address A_K. Alice uses the address A_K as the input address and A_s as the output address to construct a special transaction and publish this special transaction to the Bitcoin blockchain network. After A_s receives the amount as the output address because there is no corresponding private key, a UTXO that can never be spent will be generated, and the bitcoin will be burned and cannot be used. Finally, return the ID of the transaction is $T_sID\_T_s$. The secret information embedding the algorithm is shown in Algorithm 2.

5.3. Extract Secret Information. Alice successfully broadcasts a special transaction with a special address to the Bitcoin blockchain network, and Bob, as the receiver, needs to extract the secret information quickly and accurately. Like Alice, Bob has the key K and the symmetric key k negotiated under the chain and related algorithms and communication mechanisms used. Bob uses the key K as the private key to generate the address A_K through the function of bitcoin address generation and then obtains the transaction information related to this address by finding address A_K. Since Alice generated the address A_K in the same way and constructed the transaction T_s with A_K as the input address and A_s as the input address, the transaction with A_s as the input can be determined as a special transaction T_s containing secret information. The receiver can obtain the information of T_s by calling the API to query the address A_K, including transaction ID and the public key hash of the transaction output address. The first 128 bits of the public key hash of the output address are the ciphertext c, and the secret information m is obtained by decrypting c through the symmetric key k. The secret information extraction algorithm is shown in Algorithm 3.

5.4. Update of Key and Address. After a covert communication is over, Alice and Bob need to update the key and the input address to ensure that the address used is different each time. The update of the key ensures the security of the entire communication. Alice and Bob have the same key update mechanism and address update mechanism, and both have the key K and the ID_T_s of the last transaction of the communication. The communication parties hash the string formed by the key K and ID_T_s and use the obtained hash value as the new key K. The first 128 bits of the hash value obtained by hashing the key K are used as the new symmetric key k. After each communication, the key K and symmetric key K required for the subsequent communication are obtained through K. In this way, each bitcoin address A_K generated by K can be linked together. Alice and Bob can quickly identify special transactions through A_K. In the traditional network, overhead refers to some control information in the frame structure to ensure the completion of communication. We rely on key K and the transaction ID to update the key. Both parties share the key K, and the transaction ID exists in the bitcoin network. The transaction ID is generated by transaction information to be regarded as the overhead of the Bitcoin system. The hash operation for the key update is performed locally by the sender and receiver without additional overhead. The length of K is 256 bits, so the exhaustive cracking needs 2^{256} operations, which is computationally infeasible. The length of key K of the symmetric encryption algorithm is 128 bits, so the exhaustive cracking needs 2^{128} operations. At present, it has not been found that this algorithm has an effective vulnerability to reduce the key strength. The update of the key and address algorithm is shown in Algorithm 4.

5.5. Transaction Withdrawal Mechanism. The address generated by the fake public key hash does not have the corresponding private key, so a UTXO that can never be spent will be generated. Suppose we do not consider the negative impact of UTXO that can never be spent on Bitcoin. In this case, the risk of being suspected of special transactions will gradually increase over time, and on-chain data analysts will more and more easily detect it. The number of bitcoins burned will also affect the concealment of special transactions.

If we consider the impact of the UTXO that can never be spent on Bitcoin, we can use the transaction withdrawal

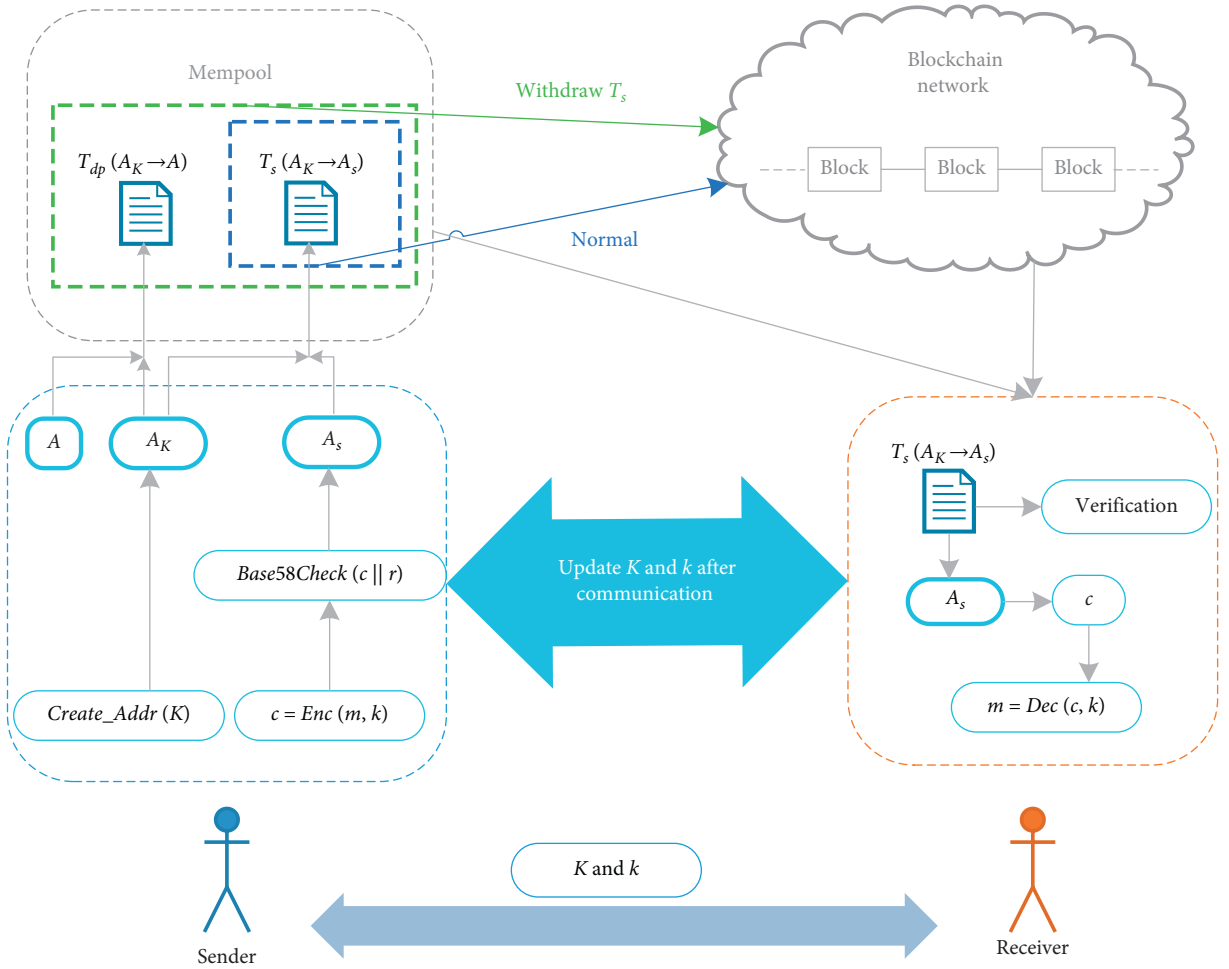

Figure 3: The model of our proposed scheme.

Table 1: Notations.

Notations	Description
M	Secret information
K	Key (as the private key of the input address)
K	Symmetric key
C	Secret information ciphertext
R	Random number
$\|\|$	The operator to concatenate two strings
$Hash\_Pk$	Public key hash
$Hash\_Pk_s$	Special public key hash (embedded information)
A	Normal address
A_s	Special address (as output address)
A_K	The address generated with the key K as the private key (as the input address)
T_s	Special transaction
Fee_s	The fee of special transaction
T_{dp}	The transaction that causes double-spend conflict
Fee_{dp}	The fee of transaction that causes double-spend conflict
ID_T_s	The ID of special transaction
ID_T_{dp}	The ID of transaction that causes double-spend conflict
Create_Addr	Function of generation bitcoin address
Base58Check	Generate a bitcoin address from the public key hash
Create_T_Broadcast	Function of create and broadcast transactions in bitcoin
GetTransaction	The blockchain API, get transaction information related to the specified address

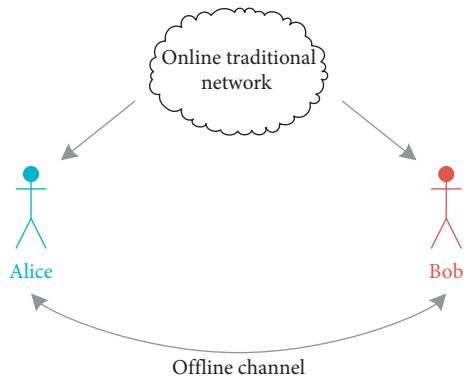

Figure 4: Exchange key and communication-related information.

mechanism based on Bitcoin that cannot be double-spent. After the special transaction T_s is broadcast by Alice, it needs to wait to be packaged into the block. Before the special transaction T_s is packaged into the block, Bob queries the address A_K to obtain the relevant information of the T_s, obtains the secret information m, and completes the covert communication. Before the special transaction T_s is packaged into the block, Alice constructs a transaction T_{dp} that causes double-spend conflict with T_s and broadcasts it and sets the fee of T_{dp} to be higher than the fee of T_s. After T_{dp} is broadcast, the transaction T_s will be cleared due to the low handling fee and cannot be packaged in the blockchain, which is equivalent to the transaction T_s being withdrawn. In this way, the UTXO that can never be spent will not be generated, and T_s will not be at risk of being suspected over time. Because Ts is withdrawn, bitcoin will not be transferred to address A_s, and the amount transferred to address A_s can be any large or small amount that fits the actual, thereby reducing the risk of suspected tracking. The transaction withdrawal algorithm is shown in Algorithm 5.

Algorithm 5 avoids generating UTXO that can never be spent and will not impact the embedding, extraction, update of key, and address in the scheme.

5.6. Scheme Analysis. In this scheme, the public key hash field in bitcoin is used as the carrier of secret information, which has high information capacity, and the encryption algorithm is used to make the carrier field unable to be distinguished from the normal field. At the same time, the carrier field does not contain any abnormal characteristics. By setting the shared key, the problem of key update and difficult transaction identification is solved, and the key update and identification transaction can be carried out safely and quickly. The mechanism of Bitcoin that can never be double-spent is used to avoid the Pay-To-Fake-Public-Key-Hash to generate a UTXO that can never be spent so that bitcoin will no longer be burned and will not cause a burden for miners. The receiver can also extract secret information safely and reliably.

This article constructs two adversary attack models, namely, off-chain traffic monitors and on-chain data analysts. When the communication parties negotiate information off the chain, the off-chain network monitor monitors its traffic. On-chain data analysts obtain valuable information by analyzing the format characteristics and content of the data on the chain and the correlation between various data and monitor data on-chain behaviors, covert communication behaviors, or some illegal behaviors. We combined two adversary attack models to analyze the scheme. All data in the public blockchain is transparent, which impacts the establishment of covert channels, but the public blockchain has a large amount of data. From another point of view, it constructs a natural umbrella for secret information, which makes it difficult for data analysts on the chain to determine the transaction of hidden secret information. For secure covert communication on the blockchain, it is necessary to ensure the anonymity of both sides of the communication, the concealment of carrier information and identification transaction mechanism, and the unbreakable security of the communication. The following will be analyzed from these three aspects.

5.6.1. Anonymity. During the off-chain negotiation process, Alice and Bob may be monitored by off-chain traffic monitors, exposing the communication information between the two parties and failing to build a secure covert channel. In this scheme, the off-chain information negotiation is only conducted once, and the negotiation is conducted under sufficiently secure conditions to ensure the safe conduct of preparations before communication and the anonymity of the communicating parties. Bitcoin itself has anonymity, but it does not have absolute anonymity, because the transactions in Bitcoin are all public. On-chain data analysts can obtain the true identity of addresses by analyzing transaction data. Therefore, the repeated use of the same address for transactions on the Bitcoin network is more likely to reveal the true identity of the address. When building transactions on the chain for communication, we make each address is used only once in the communication, and on-chain data analysts cannot infer the true identity of the address through the address that has only constructed one transaction. It should be noted that the address as the input address must have enough bitcoins. The amount source can be obtained through coin shuffle to ensure the nontraceability of the specific source of the digital currency.

5.6.2. Concealment. Concealment includes the concealment of secret information carriers and the concealment of transaction identification mechanisms. In this paper, the carrier for hiding the secret information is the public key hash field that generates the bitcoin address. The public key hash *Hash_Pk* that generates a normal address can be regarded as a 160-bit hexadecimal random number in form and content. The ciphertext c is generated after the secret information m is encrypted, equivalent to a 128-bit hexadecimal random number in form and content. Ciphertext c and random number r are combined into a special public key hash $Hash\_Pk_s$. It is indistinguishable between *Hash_Pk* and $Hash\_Pk_s$ in form and content. The addresses A and A_s

Input: Key K, symmetric key k, secret information m.
Output: The ID of the special transaction ID_T_s.
(1) $K \longleftarrow \{0,1\}^{256}$;
(2) $k \longleftarrow \{0,1\}^{128}$; /$*K$ and k are negotiated in advance $*$/
(3) $r \longleftarrow \{0,1\}^{32}$; /$*$Random generation$*$/
(4) $c \longleftarrow \{0,1\}^{128} \leftarrow \mathrm{Enc}(m,k)$;
(5) $\mathrm{Hash\_Pk}_s \longleftarrow c \| r$;
(6) $A_s \longleftarrow$ Base 58 Check $(\mathrm{Hash\_Pk}_s)$;
(7) $A_K \longleftarrow \mathrm{Create\_Addr}(K)$;
(8) $T_s \longleftarrow \mathrm{Create\_}T\mathrm{\_Broadcast}(A_K, A_s)$;
(9) $\mathrm{ID\_}T_s \longleftarrow T_s$;
(10) return ID_T_s;
(11) end;

Algorithm 2: Secret information embedding.

Input: Key K, symmetric key k.
Output: Secret information m, the ID of the special transaction ID_T_s.
(1) $K \longleftarrow \{0,1\}^{256}$;
(2) $k \longleftarrow \{0,1\}^{128}$; /$*K$ and k are negotiated in advance $*$/
(3) $A_K \longleftarrow \mathrm{Create\_Addr}(K)$;
(4) $T_s \longleftarrow$ Get Transaction (A_K);
(5) $\mathrm{Hash\_Pk}_s \longleftarrow T_s$;
(6) $\mathrm{ID\_}T_s \longleftarrow T_s$;
(7) $c \longleftarrow$ Select first 128 bits of $\mathrm{Hash\_Pk}_s$;
(8) $m \longleftarrow \mathrm{Dec}(c,k)$;
(9) return m, ID_T_s;
(10) end;

Algorithm 3: Secret information extraction algorithm.

Input: Key K, the ID of the special transaction ID_T_s.
Output: Key K, symmetric key k, the address generated with the key K as the private key A_K.
(1) $K \longleftarrow \{0,1\}^{256} \longleftarrow \mathrm{SHA\ 256}(K \| \mathrm{ID\_}T_s)$;
(2) $k \longleftarrow \{0,1\}^{256} \longleftarrow$ Select the first 128 bits of SHA 256 (K);
(3) $A_K \longleftarrow \mathrm{Create\_Addr}(K)$;
(4) return K, k, A_K;
(5) end;

Algorithm 4: Update of key and address algorithm.

generated by *Hash_Pk* and $Hash\_Pk_s$ are also indistinguishable in form and content. On-chain data analysts cannot identify the special public key hash and the special address by analyzing the public key hash field and the address field. The transactions T and T_s constructed with A and A_s as outputs are indistinguishable within a certain period. However, we took advantage of the feature of Bitcoin that cannot be double-spent to withdraw the special transaction. In this way, the Bitcoin network will not generate UTXO that can never be spent, and no one can identify whether a special transaction contains communication behavior in polynomial time. So, the carrier in our scheme has strong concealment.

The transaction identification mechanism uses a hash algorithm. If the adversary does not have the key K, according to the features of the hash algorithm, no one can determine the address A_K. Therefore, the transaction identification mechanism in our scheme also has strong concealment. If the attacker guesses a special address out of thin air, the probability of getting a correct special address is $1/2^{256}$, which will be infeasible. If the attacker guesses that the address appearing in the bitcoin network is a special address, and the number of addresses of all transactions in a period is n, the probability of getting a correct special address is $1/n$. Therefore, by guessing the special address in the

Input: The fee of special transaction Fee_s, the address generated with the key K as the private key A_K, normal address A.
Output: The transaction that causes double-spend conflict T_{dp}.
(1) After T_s is broadcast;
(2) $\text{Fee}_{dp} \longleftarrow 0$;
(3) While $\text{Fee}_{dp} \leq \text{Fee}_s$;
(4) Fee_{dp} ++;
(5) End While;
(6) $T_{dp} \longleftarrow \text{Create\_}T\text{\_Brodcast}(A_K, A, \text{Fee}_{dp})$;
(7) end;

Algorithm 5: Transaction withdrawal algorithm.

communication scene, our proposed model will not compromise.

5.6.3. Security. The security of communication is guaranteed by the key K and extends to all subsequent communications. The risk of exposure of the key K exists in the initial off-chain negotiation. It is essential to ensure the security of communication during the off-chain negotiation. After the communication parties Alice and Bob complete the off-chain negotiation, the preservation of the key K is equivalent to the preservation of the Bitcoin private key. There are many safe preservation methods, and the key K can be stored safely. The symmetric key k is obtained by hashing the key K and the hash of the special transaction. The security of K ensures the security of k. Because the hash function is irreversible, even if the symmetric key k is exposed or the encryption algorithm is cracked, it will only affect one communication. As long as the security of K is ensured, the communication security before or after will not be threatened.

In addition, in our proposed scheme, both parties have the key K, and the receiver can steal the sender's bitcoin through K, but this also means that both parties become distrustful of each other, and covert communication will become meaningless. In one-to-one communication, the betrayal of the receiver is generally not considered, and the problem of the traitor usually appears in group communication of many members. Therefore, in our scheme, both parties share the key k, which will not affect the security of communication, and k enables the receiver to verify the correctness of the transaction quickly. The private key is used at both ends, and no extra overhead is generated. Each time the two parties make covert communication, they only need to consume a handling fee, which is necessary for blockchain transactions and will not cause significant economic consumption.

6. Experiment and Comparative Analysis

This section tested the scheme using a real Bitcoin network, completed the experiment, and compared the scheme with other schemes.

6.1. Experiment on the Bitcoin Network. We use the experimental environment shown in Table 2 for experiments. The experiment uses the general performance laptop, and the Bitcoin client used is Electrum. Electrum is a SPV (Simplified Payment Verification) wallet that can realize the transaction transfer function on the chain without storing all bitcoin block data. Encryption algorithms and bitcoin address generation algorithms supporting the scheme are written in Python.

The experiment is carried out in the open real Bitcoin network. Anyone can easily verify the authenticity of the experiment and data, and better understand the functional characteristics of the scheme. For different scenarios, we set up two sets of experiments considering the problem of whether UTXO that can never be spent is generated. The secret information passed in the two experiments is "Blockchain," and different key K and symmetric key k are used.

Experiment 1. We construct a transaction using the method of Pay-To-Fake-Public-Key-Hash, and the public key hash of the output address of the transaction hides the secret information, "Blockchain." This experiment is suitable for covert communication where the effective time of communication information is relatively short and does not consider the adverse effects on Bitcoin. Table 3 shows the parameters of Experiment 1.

Firstly, the communication parties negotiate the information under the chain, such as key K and symmetric key k. The sender operates laptop A, uses the symmetric key k to encrypt the "Blockchain" to obtain the ciphertext c, and generates the corresponding special address A_s from $c||r$. Use the key K to generate the address A_K, construct and broadcast the special transaction T_s with A_K as the input address and A_s as the output address. The receiver uses the laptop B to call the API [32] to query the address A_K, and obtains transactions related to the address A_K. Because each address is used only once, the recipient can quickly and accurately identify the special transaction T_s. Use the symmetric key k to decrypt the first 128 bits of the public key hash of A_s in Ts to obtain the secret information m. After the communication is over, the two parties will update the key and address through the update mechanism. The output obtained by SHA256 hashing $K||ID\_T_s$ is used as the new key K, and the first 128 bits of the output obtained by SHA256 hashing K are used as the new symmetric key k. Table 4 shows the updated key and addresses of Experiment 1. Experiment 1 will generate a UTXO that can never be spent.

Table 2: Experimental environment parameters.

Name	Configuration/Version	Number
Laptop A	Intel(R)Core (TM)i5-5200@2.20 GHz/RAM 8G	1
Laptop B	Intel(R)Core (TM)i5-5200@2.20 GHz/RAM 8G	1
Operating system	Windows 10	—
Electrum	4.0.9	—

Table 3: Experimental parameters (Experiment 1).

Notations	Description	Content	Length
K	Key	0x77832f538aaaf37568d33979dafdaa680320beeaff668d1f6f92909b04584b9d	256 bits
k	Symmetric key	0x64306534323833336335646664353437	128 bits
m	Secret information	Blockchain	80 bits
c	Secret information ciphertext	0x8457d9399440782f6fa421fd542e7d10	128 bits
r	Random number	0xf0a1b49d	32 bits
$Hash\_Pk_s$	Special public key hash	0x8457d9399440782f6fa421fd542e7d10f0a1b49d	160 bits
A_s	Special address (as output address)	1D4mS9QBj8TwVxJjSUSVGrZVr2GFnU3EHa	34 bytes
A_K	The address generated with the key K as the private key (as the input address)	16kETg9eyFCGXddjZJkBb6LjAGXXP3oSTt	34 bytes
ID_T_s	The ID of special transaction	0x35b1b6dd6eecd67716beb4eec02c9fe0305b047f57a99d7c1d4ce4a869059c5f	64 bytes

Experiment 2. The steps of this set of experiments are the same as those of Experiment 1. The only difference is that it will not generate a UTXO that can never be spent. After the sender uses laptop A to construct a special transaction T_s and broadcast it, construct transaction T_{dp} that causes double-spend conflict and broadcast it before T_s is packaged into the block. The two transactions will conflict. Transaction T_{dp} uses A_K as the input address and A as the output address. In the experiment, the fee of T_s is 3.016 sat/byte, and the fee of T_{dp} is 10.052/byte. Bitcoin miners give priority to the transaction T_{dp} with the high fee to be packaged on the chain, and the transaction T_s will be withdrawn. In the case of keeping Fee_s always greater than Fee_{dp}, their value can be changed appropriately according to the actual situation. Table 5 shows the parameters of Experiment 2.

In Experiment 2, on-chain data analysts cannot judge whether the public key hash of the output address A_s is a fake public key hash according to the length of time, which ensures the long-term concealment of the secret information. After withdrawing the special transaction T_s, the receiver can obtain the information of T_s by calling the API [33] to query the address A_K. All the information of T_s can be obtained by querying ID_T_s on the website [34]. The special transaction T_s is not recorded on the blockchain, some nodes will continue to save this transaction, and some nodes will discard this transaction. The attacker can intercept this transaction, possibly disrupting the communication channel by modifying the transaction. However, the receiver holds the private key of the input address, which can easily verify the transaction's authenticity. Table 6 shows the updated key and addresses of Experiment 1.

The experiment considered short-term and long-term secret information. Aiming at the two types of information and considering the impact on Bitcoin, two experimental schemes are proposed. Experiment 2 is an improvement scheme of Experiment 1. This scheme uses the public key hash field of the address to hide secret information, and its maximum capacity can reach 120 bits. From the perspective of embedded capacity, constructing a transaction can meet the capacity requirements of regular covert communications. The indistinguishability of the public key hash field and the feature of Bitcoin that cannot be double-spent together ensure the concealment of secret information. The private key is derived from the key hash, and then the address is generated. The method of relying on the address to identify the transaction avoids heavy screening tasks and dramatically improves the identification efficiency of the transaction. Table 7 shows the time spent in all additional operations during the communication process. The additional time used by the receiver is about 9 milliseconds. After the sender broadcasts a special transaction and the corresponding blockchain node receives the transaction information, the time to obtain the transaction is about 3 seconds, and the time for the receiver to obtain the secret information is about 44 milliseconds. It can be seen that our scheme spends very little time in the entire communication process, and the entire communication can be completed quickly.

6.2. Scheme Comparison and Analysis. This section compares the previous blockchain covert communication scheme with the scheme in this article. Table 8 shows the

Table 4: Updated key and address (Experiment 1).

Notations	Description	Content	Length
K	Key	0xedb53cfdd24edb48cfc9a58ad4c2e9a5d4e905358047383993fbdb525952c178	256 bits
k	Symmetric key	0x79b946d16b547aa4399c85ff409e2961	128 bits
A_K	The address generated with the key K as the private key (as the input address)	1Ke9bGkRzpHe5AE6Ei18bULT95eCVehFfM	34 bytes

Table 5: Experimental parameters (Experiment 2).

Notations	Description	Content	Length
K	Key	0xa00ccf6f1e9dad5fb65989b1f5697e877af0b86a1b1bf413eece4911f05d439c	256 bits
k	Symmetric key	0x36633565646435343236646163653961	128 bits
m	Secret information	Blockchain	80 bits
c	Secret information ciphertext	0xeb82259e8576a117e4e31d6ef5bd2dcf	128 bits
r	Random number	0xd304321a	32 bits
$Hash\_Pk_s$	Special public key hash	0xeb82259e8576a117e4e31d6ef5bd2dcfd304321a	160 bits
A_s	Special address (as output address)	1NUFhrEGWP3ZnZx8a1D7vwv7MubJH5tNTG	34 bytes
A_K	The address generated with the key K as the private key (as the input address)	1FGNgSaCCDqwFNffNkPXnBeCSXcWP6WVrC	34 bytes
ID_T_s	The hash of special transaction	0x97e794607851d0d752adc5459516a7be3d36da8d4dc62d23c38f4e02237a11a2	64 bytes
A	Normal address	1PSL1uSGs3LieZ5iYKRbtQFTpH2KCxgoGc	34 bytes
Fee_s	The fee of special transaction	3.016 sat/byte	—
Fee_{dp}	The fee of transaction that causes double-spend conflict	10.052 sat/byte	—
ID_T_{dp}	The ID of transaction that causes double-spend conflict	0x4404f05fa0772d7d78a804faed33760c2d9466c40a49dda3d1b4215507687834	64 bytes

Table 6: Updated key and addresses (Experiment 2).

Notations	Description	Content	Length
K	Key	0x5a4d8d037aeaff2ea3b03d5f7bbf1e689759dff94f687a8059e336e8251c6ca2	256 bits
k	Symmetric key	0x1757ad095901f8749de10970211fda6a	128 bits
A_K	The address generated with the key K as the private key (as the input address)	12QrykwTDsjudiysdcTV6XNfLhEaCMJVzG	34 bytes

comparison between this scheme and other blockchain covert communication schemes in terms of embedding mechanisms.

It can be seen from the table that the information capacity of this scheme and other schemes are at an intermediate level in terms of information capacity. However, our scheme has stronger concealment and higher efficiency. Because everything will become meaningless if the information is exposed, we take concealment as the most important indicator. When embedding secret information, the concealment and information capacity combined with the method of embedding secret information can reflect the scheme's efficiency. Embedding secret information in the amount field and OP_RETURN field can easily make the field special and weaken the concealment of the carrier. The three schemes "BLOOCE," "V-BLOOCE," and "BLOOCE+" use the least significant bits to hide information and have strong concealment.

Table 7: Time costs.

Description	Time Costs
Generate random number (ms)	0.019
Encryption (ms)	1.077
Decryption (ms)	0.999
Generate bitcoin address through private key (ms)	5.988
Generate bitcoin address through public key hash (ms)	1.008
Call API to get transaction T_s (s)	2.919389
Call API to get withdrawn transaction T_s (s)	3.215613
Update key and address (ms)	1.002
Verify signature (ms)	39.042

Table 8: Embedding mechanism.

Scheme	Carrier	Capacity (single transaction)	Concealment
Amount field [17]	Amount field	16 bits	Weak
BLOOCE [18]	Least significant bit of address	1 bit	Strong
Kleptography [21]	OP_RETURN	80 bytes	Weak
V-BLOOCE [22]	The last character of the address	$\log_2 58$ bits	Strong
BLOOCE+ [23]	The last α bits of the address	α bits	Strong
Ours	Public key hash	120 bits	Strong

Table 9: Transaction identification mechanism.

Identification mechanism	Concealment	Efficiency
Fixed address [18]	Weak	High
Update address in the channel [19]	Strong	High
Kleptography algorithm (fixed address) [21]	Weak	Low
Dynamic label (based on improved DGA) [27]	Strong	Middle
Dynamic label (based on HMAC) [28]	Strong	Middle
Ours	Strong	High

Assume that the size of the secret information that needs to be transmitted is 120 bits. The "BLOOCE" scheme requires the construction of 120 transactions, each corresponding to a bit, and in order to maintain concealment, each transaction needs to be dispersed in different blocks. According to the principle that Bitcoin generates a block in 10 minutes, it takes 1200 minutes to embed 120 bits of information. The "V-BLOOCE" scheme requires about 21 transactions to be constructed and takes 150 minutes. Our scheme only needs to construct a transaction, and the receiver can receive it after the transaction is broadcast. For the "BLOOCE+" scheme, assuming that α is 120, only one transaction needs to be constructed, but a substantial addition of calculation is required. If the hash function is in an ideal state, a result needs to be calculated from a uniformly random bit sequence $(0,1)^{\alpha}$. Suppose the computational complexity of generating an address is C_{Gen}, and generating a specific address is $2^{\alpha}C_{\mathrm{Gen}}$. It takes about 6 milliseconds to generate a bitcoin address using python, so the time to generate a specific address is about $6 * 2^{\alpha}$ milliseconds. However, the time used to embed secret information in our scheme is only about 9 milliseconds.

We have compared the identification mechanisms of special transactions containing secret information. Table 9 shows the comparison results. The identification efficiency of the special transaction identification mechanism is reflected in the time it takes for the receiver to identify the special transaction after the special transaction is broadcast. In addition, the special transaction identification mechanism also needs to be very concealed to ensure that no one except the receiver can identify the special transaction.

Using a fixed address to identify a special transaction is the most efficient identification method. It is easy to get all the information of a special transaction by calling the API, but this will inevitably re-use the address and increase the risk of channel exposure. Updating the address in the channel will occupy the channel capacity. Using the kleptography algorithm to identify special transactions requires calculation and analysis of the signatures of all new transactions, equivalent to traversing each transaction. Using dynamic labels (based on improved DGA) to identify special transactions requires traversing all newly OP_RETURN fields. Using dynamic tags (based on HMAC) to identify special transactions requires traversing all newly generated blocks. The above three methods all need to undertake cumbersome screening tasks. In the experiment of our scheme, all the information of the special transaction can be obtained by calling the API to query the address A_K. Our

special transaction identification mechanism is similar to a fixed address to identify special transactions and has very high identification efficiency. Moreover, the concealment of the identification mechanism of our scheme is guaranteed by the key K, and no one can identify the special transaction without the key K.

7. Conclusion and Future Work

This paper proposes a covert communication scheme based on the bitcoin transaction mechanism, which uses a symmetric key to ensure the security and concealment of secret information. The private key is used to ensure that the receiver can quickly and accurately identify a special transaction that contains secret information. Without a private key, no adversary can determine a special transaction that contains secret information. The iterative hash method updates the key to ensure that the communication address is different each time. It is taking advantage of the feature of Bitcoin that cannot be double-spend to avoid the Pay-To-Fake-Public-Key-Hash method generating UTXO that can never be spent. We tested it on the real Bitcoin network and proved that our scheme is very safe and efficient. Anyone can verify the authenticity of the experiment based on the data we provide.

In the scheme proposed in this paper, the private key is in the hands of both communicating parties, which involves the problem of the receiver's reliability. In order to avoid some possible problems, a special secure transaction identification mechanism can be studied in which the receiver cannot obtain the private key. Whether there is any particularity in the withdrawn bitcoin transaction should be further discussed. The withdrawal of the transaction needs to be carried out when the Bitcoin network is congested. Otherwise, it may not be possible to withdraw it. In future work, we will explore the scalability of this scheme and continue to study other more efficient and secure covert communication schemes in the Bitcoin blockchain. In addition, with the gradual popularization of blockchain technology, combining multiple public blockchains with building a covert channel is also the future direction of development.

Data Availability

All the data in this study were taken from experimental data statistics.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by National Nature Science Foundation of China (62072290), the Key Research and Development Program of Shandong Province (2019GNC106027 and 2019JZZY010134), and the Natural Science Foundation of Shandong Province (ZR2020MF058 and ZR2021MF118).

References

[1] B. W. Lampson, "A note on the confinement problem," *Communications of the ACM*, vol. 16, no. 10, pp. 613–615, 1973.

[2] R. Anderson, "Stretching the limits of steganography," in *Proceedings of the International Workshop on Information Hiding*, pp. 39–48, Cambridge, UK, June 1996.

[3] P. Malaichamy, S. Thangavel, V. Sonai, and G. Gopal, "A novel encryption of text messages using two fold approach," *Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Sciences*, vol. 10, no. 6, pp. 1106–1112, 2021.

[4] G. G. Deverajan, V. Muthukumaran, C. H. Hsu, M. Karuppiah, Y. C. Chung, and Y. H. Chen, "Public key encryption with equality test for industrial internet of things system in cloud computing," *Transactions on Emerging Telecommunications Technologies*, Article ID e4202, 2021.

[5] S. Palanisamy, R. Somula, and G. G. Deverajan, "Communication trust and energy-aware routing protocol for WSN using D-S theory," *International Journal of Grid and High Performance Computing*, vol. 13, no. 4, pp. 24–36, 2021.

[6] T. G. Handel and M. T. Sandford, "Hiding data in the OSI network model," in *Proceedings of the International Workshop on Information Hiding*, pp. 23–38, Cambridge, UK, May 1996.

[7] J. Millen, "20 Years of covert channel modeling and analysis," in *Proceedings of the 1999 IEEE Symposium on Security and Privacy (Cat. No.99CB36344)*, pp. 113-114, Oakland, CA, USA, May 1999.

[8] S. Lu, Z. Chen, G. Fu, and Q. Li, "A novel timing-based network covert channel detection method," in *Proceedings of the 2019 International Conference on Artificial Intelligence Technologies and Applications*, Article ID 012050, Qingdao, China, July 2019.

[9] J. Xie, Y. Chen, L. Wang, and Z. Wang, "A network covert timing channel detection method based on threshold secret sharing," *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, Article ID e3781, 2020.

[10] K. Marimuthu, D. G. Gopal, S. Aditya, and V. Mittal, "Cryptanalysis of oPass," in *Proceedings of the 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, pp. 329–334, Ramanatapuran, India, May 2014.

[11] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," *Decentralized Business Review*, Article ID 21260, 2008.

[12] Y. Wang, Y. Wang, Z. Wang, G. Yang, and X. Yu, "Research cooperations of blockchain: toward the view of complexity network," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–14, 2020.

[13] K. Marimuthu, D. G. Gopal, G. Malik, and P. Boominathan, "A secured cloud system and log records based on 2LE," *International Journal of Applied Engineering Research*, vol. 9, no. 20, pp. 7435–7451, 2014.

[14] T. Li, Y. Chen, Y. Wang et al., "Rational protocols and attacks in blockchain system," *Security and Communication Networks*, vol. 202011 pages, Article ID 9856969, 2020.

[15] P. Li, K. Li, Y. Wang et al., "A systematic mapping study for blockchain based on complex network," *Concurrency and Computation: Practice and Experience*, Article ID e5712, 2020.

[16] J. Warren, "Bitmessage: a peer-to-peer message authentication and delivery system," *White Paper*, 2012.

[17] M. D. Sleiman, A. P. Lauf, and R. Yampolskiy, "Bitcoin message: data insertion on a proof-of-work cryptocurrency system," in

Proceedings of the 2015 International Conference on Cyberworlds (CW), pp. 332–336, Visby, Sweden, October 2015.

[18] J. Partala, "Provably secure covert communication on blockchain," *Cryptography*, vol. 2, no. 3, p. 18, 2018.

[19] D. Frkat, R. Annessi, and T. Zseby, "ChainChannels: private botnet communication over public blockchains," in *Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 1244–1252, Halifax, Nova Scotia, Canada, July 2018.

[20] Y. Li, L. Ding, J. Wu, Q. Cui, X. Liu, and B. Guan, "Research on a new network covert channel model in blockchain environment," *Journal on Communications*, vol. 40, no. 5, pp. 67-68, 2019.

[21] F. Gao, L. Zhu, K. Gai, C. Zhang, and S. Liu, "Achieving a covert channel over an open blockchain network," *IEEE Network*, vol. 34, no. 2, pp. 6–13, 2020.

[22] L. Zhang, Z. Zhang, W. Wang, and R. Waqas, "A covert communication method using special bitcoin addresses generated by vanitygen," *Computers, Materials & Continua*, vol. 65, no. 1, pp. 495–510, 2020.

[23] S. Song and W. Peng, "BLOCCE+: an improved blockchain-based covert communication approach," *Journal of Chongqing University of Technology (Natural Science)*, vol. 34, no. 9, pp. 238–244, 2020.

[24] L. Zhang, Z. Zhang, W. Wang, Z. Jin, Y. Su, and H. Chen, "Research on a covert communication model realized by using smart contracts in blockchain environment," *IEEE Systems Journal*, pp. 1–12, 2021.

[25] W. She, L. Huo, Z. Tian, Y. Zhuang, C. Niu, and W. Liu, "A double steganography model combining blockchain and interplanetary file system," *Peer-to-Peer Networking and Applications*, vol. 14, no. 10, pp. 1–14, 2021.

[26] Z. Guo, L. Shi, M. Xu, and H. Yin, "MRCC: a practical covert channel over Monero with provable security," *IEEE Access*, vol. 9, pp. 31816–31825, 2021.

[27] J. Tian, G. Gou, C. Liu, Y. Chen, G. Xiong, and Z. Li, "DLchain: a covert channel over blockchain based on dynamic labels," in *Proceedings of the International Conference on Information and Communications Security*, pp. 814–830, Bei Jing, China, December 2019.

[28] C. Si, F. Gao, L. Zhu et al., "Covert data transmission mechanism based on the dynamic label in the blockchain," *Journal of Xidian University*, vol. 47, no. 5, pp. 94–102, 2020.

[29] A. Sward, I. Vecna, and F. Stonedahl, "Data insertion in bitcoin's blockchain," *Ledge*, vol. 3, pp. 1–23, 2018.

[30] OKlink.com, "Transaction record," https://www.oklink.com/btc/tx/8881a937a437ff6ce83be3a89d77ea88ee12315f37f7ef0dd3742c30eef92dba.

[31] W. Chen and Z. Zheng, "Blockchain data analysis: a review of status, trend and challenges," *Journal of Computer Research and Development*, vol. 55, no. 9, pp. 1853–1870, 2018.

[32] B. Cypher, *Bitcoin API*, https://www.blockcypher.com/dev/bitcoin/#introduction.

[33] T. View, *Bitcoin API*, https://documenter.getpostman.com/view/5728777/RzZ6HfX2?version=latest#1ff89c3f-e8dc-45c6-af53-4a7a84bace9e.

[34] OKlink.com, "Transaction records," https://www.oklink.com/btc/tx/97e794607851d0d752adc5459516a7be3d36da8d4dc62d23c38f4e02237a11a2.