Research on Blockchain Anonymous Communication Based on Key Derivation

Yanxun Chen
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail: 872887719@qq.com

Pingping Liu
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail: 1341369601@qq.com

***Abstract*—With the continuous development of the Internet and communication technologies, network communication provides convenience but also brings security problems such as exposure of users' personal privacy information and theft, private tampering, and forgery of false information. Modern cryptography technology is an important safeguard against message eavesdropping and tampering, while the rapidly developing anonymous communication technology in this century makes it difficult for attackers to infer user's personal information and communication relationships. In response to the potential threats of traditional centralized systems such as central nodes being vulnerable to attacks and data storage being tampered with, this paper proposes a blockchain anonymous communication algorithm KDAC based on key derivation, which takes advantage of the decentralization, data immutability, consensus mechanism and anonymity of blockchain and combines the ECC cryptographic derivation algorithm and anonymous communication technology to realize the key-at-a-time, one-address-at-a-time The key derivation scheme ensures the message integrity and tamper-evident while effectively hiding the identity information of both communication parties. In addition, this paper also optimizes the blockchain anonymous communication system with key derivation. Users only need the initial key of blockchain nodes to join the network for communication, and the information transmission is difficult to trace based on the blockchain network, which can effectively guarantee communication security and anonymity. The experimental results show that the efficiency of the derived key algorithm is roughly in the same order of magnitude as that of the 256-bit AES symmetric encryption algorithm, which can play a better role in practical applications. On the other hand, the derived key generated based on the algorithm has complete randomness in association verification, and it is impossible to reverse the initial parameters, which can well guarantee the anonymity of user identity.**

Keywords-Blockchain; Key Derivation; Anonymous Communication; One Secret At a Time

I. INTRODUCTION

Various attacks exist in network communication at present, including illegal interception, tampering, and forged false information on the way of message transmission, which largely affects network security and makes people face great privacy threats [1]. For applying blockchain technology to anonymous communication to achieve secure transmission of information, there are few researches or technical implementation results on this topic at home and

DOI: 10.2478/ijanmc-2023-0051

abroad. Flooding algorithms such as Flooding and Epidemic are widely used in P2P networks to achieve anonymous communication, in which the uncertainty of message transmission path and the concealment of arbitrary nodes are very effective to ensure communication anonymity [2]. In 2002, Freedman designed Tarzan, a system similar to the Mix anonymous communication system but with more scalability for P2P networks [3]. In 2006, Tianbo Lu et al. combined the advantages of Crowds and Mix systems, combining layered cryptography and the idea of random forwarding of nodes in P2P networks to complete the secure and efficient anonymous communication system WonGoo [4]. Juha Partala used the existing ideal blockchain model as a basis to design a method that can securely embed steganographic messages into the blockchain and demonstrated that the system BLOCCE implemented based on this model is secure and steganographic for communication [5]. The improvement of the covert communication system BLOCCE was made at Song of Lanzhou University, which improved the overall communication efficiency as well as the continuity of the communication process [6].

This system uses the excellent features of blockchain itself to solve many problems in the field of covert communication, including anonymity, de-trust, and concealment. The covert communication simultaneously counteracts the blockchain and solves the pain points of data security and privacy of its application system. In this paper, we propose the KDAC (Key Derived Anonymous Communication) algorithm, which is a secure communication scheme based on the existing alliance chain as a platform and combined with cryptographic principles, which greatly and effectively ensures the data integrity and steganography, which is also an innovation based on the security advantages of blockchain combined with other theoretical technologies.

II. Related Job

A. Blockchain Technology

Blockchain technology, as a collection of fusion of various cutting-edge technologies, its technical details have very important practical significance and reference value for the whole digital currency system as well as for other industries. The anonymity, immutability, decentralization, and other characteristics of blockchain will also become irreplaceable advantages on certain special occasions. In the field of secure communication, the use of blockchain network can effectively hide the identity of both sides of communication, while making the communication channel untraceable. At the same time, the huge amount of users and address space of blockchain provide an excellent cover for both communicating parties, allowing users to keep changing addresses and keys during the communication without attracting the attention of attackers. The ultimate goal of this study is to be compatible with large blockchain network systems so that the communication process from the observer's perspective is no different from ordinary transactions, and the transition from transactions to communications is truly realized.

B. ECC Cryptography

The development of blockchain cannot be separated from public-key cryptography. In mainstream digital currency platforms such as Bitcoin and Ether, ECC cryptography can assume the role of a mainstay, thanks to the rigorous mathematical one-way mapping relationship of private key to public key in ECC cryptography and the significance of ESCDA algorithm to generate unforgeable signatures [7]. In the

communication model of this paper, unlike the transaction logic of digital currencies, the communication message is not completely transparent as the transaction message, making it accessible and verifiable by anyone. In addition to sending the message and generating a signature that can verify their identity, the sender needs to encrypt the message to generate a ciphertext to realize the session logic of encrypting the message signature by the sender and decrypting it by the receiver to verify it, and the elliptic curve encryption algorithm fits perfectly to the needs of this study.

C. Layered Key Technology

One of the representatives of hierarchical key technology is the HD wallet (Hierarchical Deterministic Wallet), which is a class of deterministic wallets that mainly uses derivation algorithms to derive any number of subkeys from a master key generated by a secure random number, where the algorithms have deterministic and irreversible characteristics. It has the advantages of convenient backup, safer offline storage of private keys, and convenient authority control. The layered key technology of Ethernet HD wallet can well solve the security problems such as a single key address, based on which the key derivation scheme of one key at a time and one address at a time is proposed, combined with the knowledge of ECC elliptic curve cryptography, to realize the communication between the two parties using the derived keys without exchanging keys, which further enhances the steganography of communication.

III. KDAC ALGORITHM

A. System model design

The development goal of the system is to combine the existing federated chain to establish a complete communication system. The specific communication model design will ensure that it has good message confidentiality and anonymity and that the communication is reliable and difficult to be attacked, but the actual application scenario will sacrifice a certain amount of real-time communication capability. The communication model design is shown in Fig. 1.

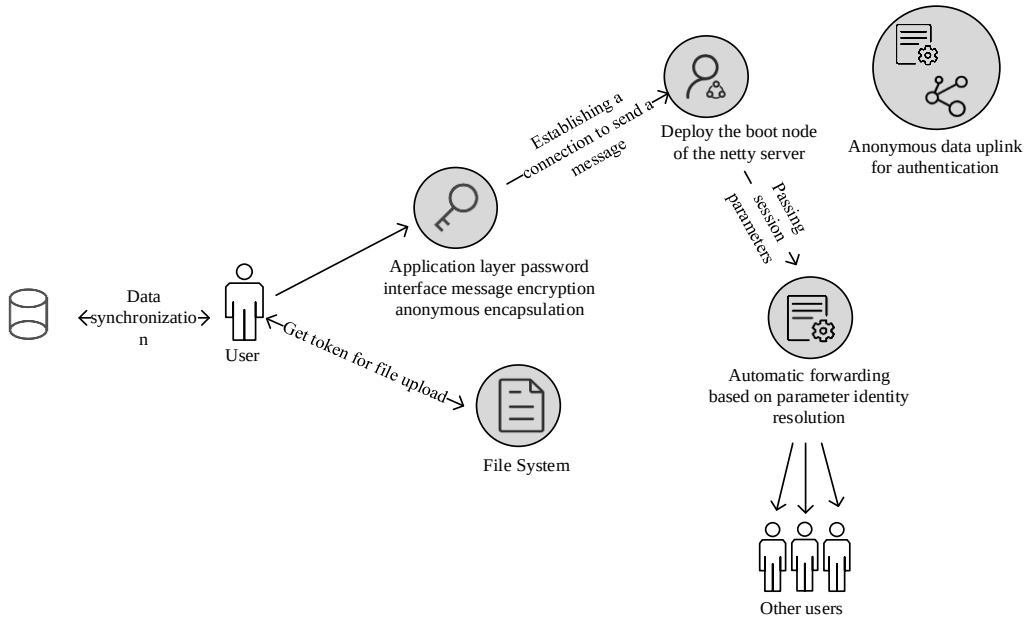

Figure 1. System communication model design

B. System architecture design

Based on the analysis of the system requirements and the initial plan of communication, the system architecture is designed as shown in Fig. 2. The system is mainly divided into two modules, the user module and the federated chain module, the user module mainly includes the application layer, algorithm layer, and storage layer, and the federated chain module is mainly divided into the communication application layer, chain consensus layer, and chain storage layer, the user interacts with the communication application layer of the existing federated chain through the application layer to achieve secure and anonymous communication.

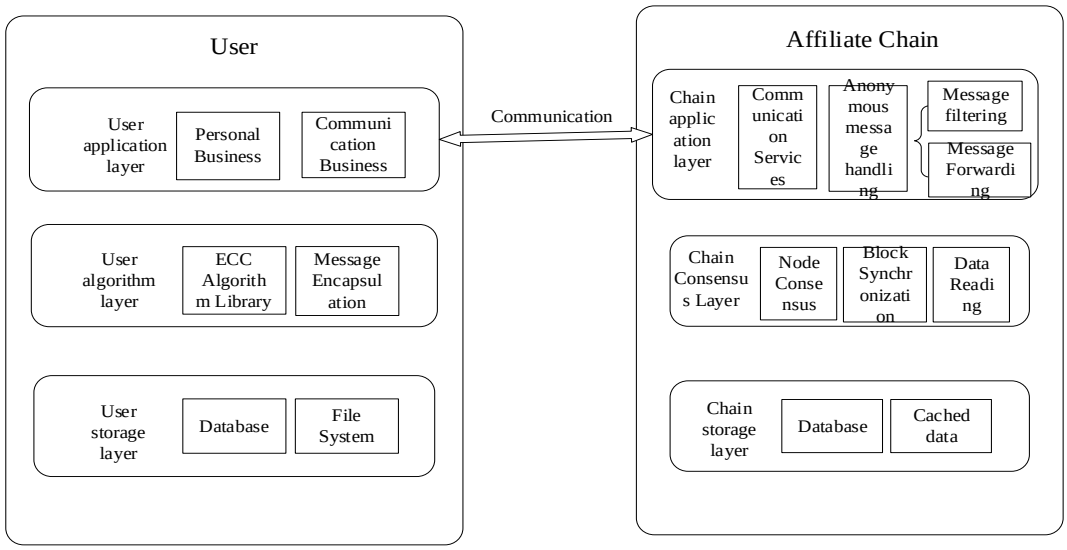

Figure 2. System layered architecture diagram

C. Block Security Synchronization

Blockchain is a distributed storage structure in terms of content storage, and the nodes in its underlying p2p network all have exactly equal power. Decentralization is a core element in blockchain, so the absence of a central server also causes the problem of an untrustworthy network environment. To ensure the consistency of blockchain data among nodes, some method is needed to solve the trust problem among nodes and to synchronize data efficiently [8]. The consensus mechanism can meet this need very well and can make the nodes cooperate in solving problems trustfully [9].

For the problem of inefficiency of the original Byzantine fault-tolerant algorithm, PBFT, the practical Byzantine fault-tolerant algorithm, is used in this paper. the PBFT algorithm is improved accordingly, which makes it possible to better solve the communication consistency problem in the non-trusted environment in practical applications.

The process of synchronizing request information in the PBFT consensus algorithm is divided into the following stages, together with the process of master node generation and the final response to the synchronization results, which are described in the following steps.

Step 1: Request: A node is selected as the master node from the network, such as node 0 in the figure;

Step 2: Pre-preparation: The customer service end starts to send specific request events, and the nodes that receive them will diffuse them in the network, then the LEADER will store the collected requests in order and broadcast them again, and then move to the next stage, as in Fig. 3 node 0 will diffuse the request messages to nodes 1, 2, and 3;

Step 3: Preparation: After each node receives the list, it generates local blocks according to the sorting, and then broadcasts to the whole network according to the hash digest of the new blocks, combined with data technology, as in Fig. 3 Node 1 broadcasts to 0, 2, 3, and Node 2 broadcasts to 0, 1, 3. Suppose Node 3 is offline for some reason and cannot broadcast;

Step 4: Confirmation: If a node receives more than 2f digests broadcast by other nodes equal to the local calculation, it continues to broadcast an acknowledgment message to the whole network;

Step 5: Response: If the master node receives a 2f+1 transfer confirmation message, it can be regarded as a successful response and can submit a new block containing the requested information to the local blockchain and state database to complete the synchronization of the blockchain.

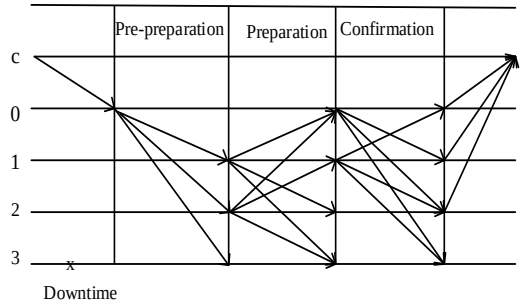

Figure 3. PBFT consensus algorithm information synchronization response process

D. KDAC Algorithm Design

In the existing scheme, the messages in the communication will be hidden by encryption at the client side to ensure the privacy of the ciphertext, but other information of the communicating parties, such as logical addresses and public keys, will still be open to others as on-chain data. To address the above problems, this paper proposes the KDAC algorithm, which in order to better utilize the blockchain network as a secure privacy channel and weaken the authority of the server to a certain extent, the federated chain also no longer forwards messages directly based on the logical address to physical address mapping relationship of the target, but is improved to send continuously updated session parameters by the receiving party to interact with the chain to obtain messages automatically. The anonymity of communication is further ensured by cryptography, which hides the identity information of the communicating parties and enables the receiver to receive messages from the federation chain server accurately, solving the problem of insufficient security and privacy of the identity information of the communicating parties.

Key Derivation Algorithm Design

The hierarchical key system in the HD wallet provides a key derivation strategy to achieve one key at a time, however, it has some limitations in the communication scheme. This system is based on the key derivation algorithm of the HD wallet technology, and the key derivation algorithm that implements a one-at-a-time account, "use-it-or-lose-it", private key opacity, and public key translucency is studied and applied to communication.

According to the key mapping principle in ECC finite cyclic group, it is easy to map from the private key to the public key, but it is impossible to push out the private key from the public key.

On this basis, the existing public key K is used as the base point of the elliptic curve, and given a new random number x as the seed, a dot product operation on the elliptic curve is done to map to the new public key K'. By the same token, as long as the "new private key" x is large enough, it is impossible to push out x from K'. The specific design of the non-negotiable one-at-a-time key is as follows.

Firstly, given an elliptic curve Ep(A, b) and its upper base point G, the order is n, and provide the private key $k \in [0, n-1]$, the formula can be obtained as follows:

$$K = kG \tag{1}$$

Then do the product operation of integer x for the public key K in equation (1), and since the group is cyclic, the new public key K' is obtained, which can be transformed into equation (2).

$$K' = xK = xkG = (kx \bmod n)G \tag{2}$$

Finally, from Equation (2), the new public key K' can also be obtained by mapping the new private key $k' = kx \bmod n$, and k' is derived by multiplying x with the integer cyclic group of k in [0, n-1]. That is, the user can derive his own new legal key pair (k', K') by x for the old key pair (k, K). As shown in Fig. 4 under this rule, $k \to K$, $k \to K'$, $k' \to K'$ are one-way mappings.

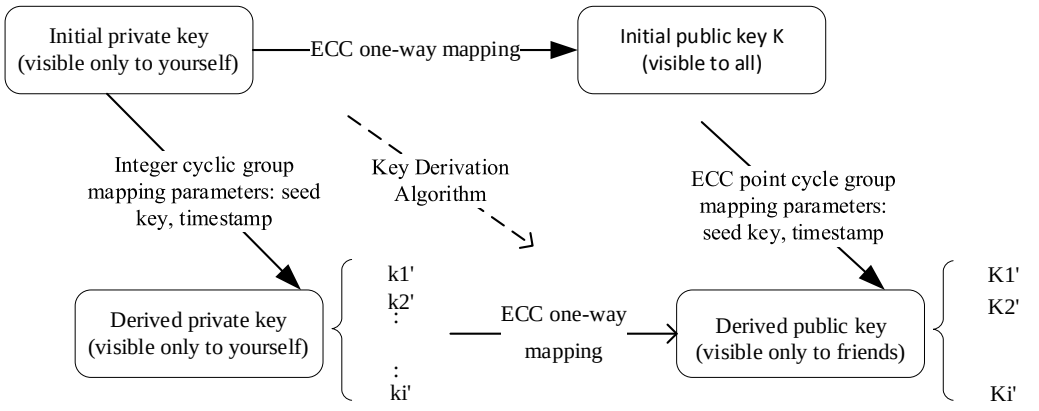

Figure 4. ECC key derivation mapping rules

In addition to keeping their private keys secret, the two communicating parties need to share and keep secret the "seed key" x in some way, so that they can each derive their own new key pair and can derive each other's new public key K', and then use the time stamp t as the variable parameter to let x change without After that, using the timestamp t as the variable parameter and letting x change without rules, we can derive a session public key that is transparent and irreversible to both parties, achieving a pseudo-random one-at-a-time effect, and not requiring a separate secure channel for key distribution negotiation, etc. The specific process is as follows.

Step 1: User A and User B initialize their key pairs, negotiate the appropriate ECC cyclic group Ep(a, b), and each shares the public key KA and KB, which cannot be changed after the initial key pair is generated;

Step 2: Users A and B each generate xA and xB using a cryptographically secure random number generator, and then transmit the shared xA and xB under a reversible public key cryptosystem and combine them as parameters using the relevant algorithms to obtain a 64-bit seed key x. The seed key is shared successfully;

Step 3: In a formal session, user A gets the current timestamp t, does a hash one-way operation on the seed key x as an argument to get xt, and then does a dot product derivation within Ep on user B's public key KB to get the other party's temporary session public key KBt;

Step 4: User A can choose to use ECDH key exchange algorithm to get $S = K_A K_{Bt}$, and xor point S to get a 256-bit integer as symmetric session key KS, and then use KS to encrypt the message, and send it to B together with timestamp t;

Step 5: User B obtains the timestamp t sent by A and generates A's temporary session public key KAt in the same way as step 3, and uses the ECDH key exchange algorithm to obtain $S = K_B K_A$, after which KS is obtained to decrypt the message, completing a one-at-a-time, key-opaque session that changes with the timestamp.

The purpose of introducing the timestamp parameter t and doing the hash operation is twofold: one is to generate different x' for each communication to achieve one-at-a-time encryption; the other is to make use of the one-way nature of the hash operation, even if the attacker obtains the derived public key of the target in some way, it cannot reverse the resolution of the corresponding seed key x, which ensures the security of key derivation. This can further increase the discrete degree of communication address and avoid insecurity in communication to a greater extent on the basis of achieving one-at-a-time encryption to enhance message opacity. The specific generation and transparency characteristics of key and address derivation are shown in Table 1.

TABLE I. ECC KEY GENERATION PROCESS AND TRANSPARENCY CHARACTERISTICS TABLE

Key Type	Generation process	Transparency
Initial private key	k	Only visible to yourself
Initial public key	K=kG	Visible to all
Derived Private Keys	K_t=k*HASH(x‖t)mod n	Only visible to yourself
Derived Public Key	K_t=K*HASH(x‖t)=k_tG	Both sides of the communication are visible
Session Key	$K_{st}=K_AK_{Bt}=k_BK_{At}$	Both sides of the communication are visible

1) Design of Cryptographic Communication with Derived Keys

The encrypted communication logic based on the key derivation algorithm is the core of this secure communication system. To ensure the confidentiality of all kinds of messages and anonymity during transmission, the following design scheme is mainly followed.

The communication message transmission flow is designed as follows

Step 1: User A sends a message to friend B and sends the message to the server.

Select the message type, if it is text then edit the text message, if it is a file it will be encrypted and uploaded to the file system to get the link when sending, the link will be sent as a text message;

Generate the B-derived public key KBt and session key Kst at the current timestamp t by using the seed key shared with B. Encrypt the

edited message (message type and message) with the session key to get the ciphertext m;

Package the ciphertext m, communication timestamp t, and derived public key KBt into a JSON file and send it to the chain server, and the sending message is finished.

Step 2: User B receives the message sent by A. The message is forwarded locally by the chain and then decrypted.

B gets the message JSON file and verifies that the message is sent by A. Extract the seed key x shared with A and generate the session key Kst under timestamp t to decrypt the message to get the plaintext M;

Get the message type of M, if it is a file type then extract the file from the file system to decrypt it, and then render the received content to the message page. Message reception is completed.

Step 3: The JSON message uploaded to the chain contains only ciphertext and derived information, so the third party cannot restore the plaintext and the initial public key information of both parties, thus achieving the effect of anonymity. The service of parsing session parameters deployed on the chain can parse the JSON messages to know the addresses of both parties and realize automatic forwarding.

Alliance chain communication guide nodes get the derived session parameter information under the corresponding timestamp through the heartbeat messages sent by all users, and store it to the cache map with the user key as the key; the messages are uploaded to the blockchain and completed consistency synchronization will be sent to the message queue for processing through asynchronous mode; the key derivation algorithm is used for efficient comparison and screening to determine the initial public key of the message recipient and map it to the specific IP address, and then forwarding can be done.

E. Federation Chain Communication Design

The code for the communication function of the federated chain is written and implemented based on Java and Netty open source framework for link-in invocation. Each time a new block is uploaded to the chain, a consensus node is selected as the master node to act as the consensus initiator node. To ensure fairness as well as stability, the polling method is used to achieve load balancing and master node election so that consensus nodes have equal opportunities to participate in the block out.

1) Alliance chain communication function module

The communication function of the federated chain mainly includes establishing and maintaining connections for online users, receiving various types of messages and automatically parsing and forwarding messages, and other processing of messages.

The message types parsed by the application layer of the federation chain mainly include heartbeat messages, request connection messages, session messages, etc. Among them, heartbeat messages are mainly used for updating user session parameters, request connection messages are used for establishing long connections with the chain and releasing key and other information, and session messages are used for formal communication, which can be used to resolve the identity and automatically forward to the target user, and to synchronize the relevant data to generate blocks that can be verified by both sides of the communication, etc. The specific process is shown in Fig. 5.

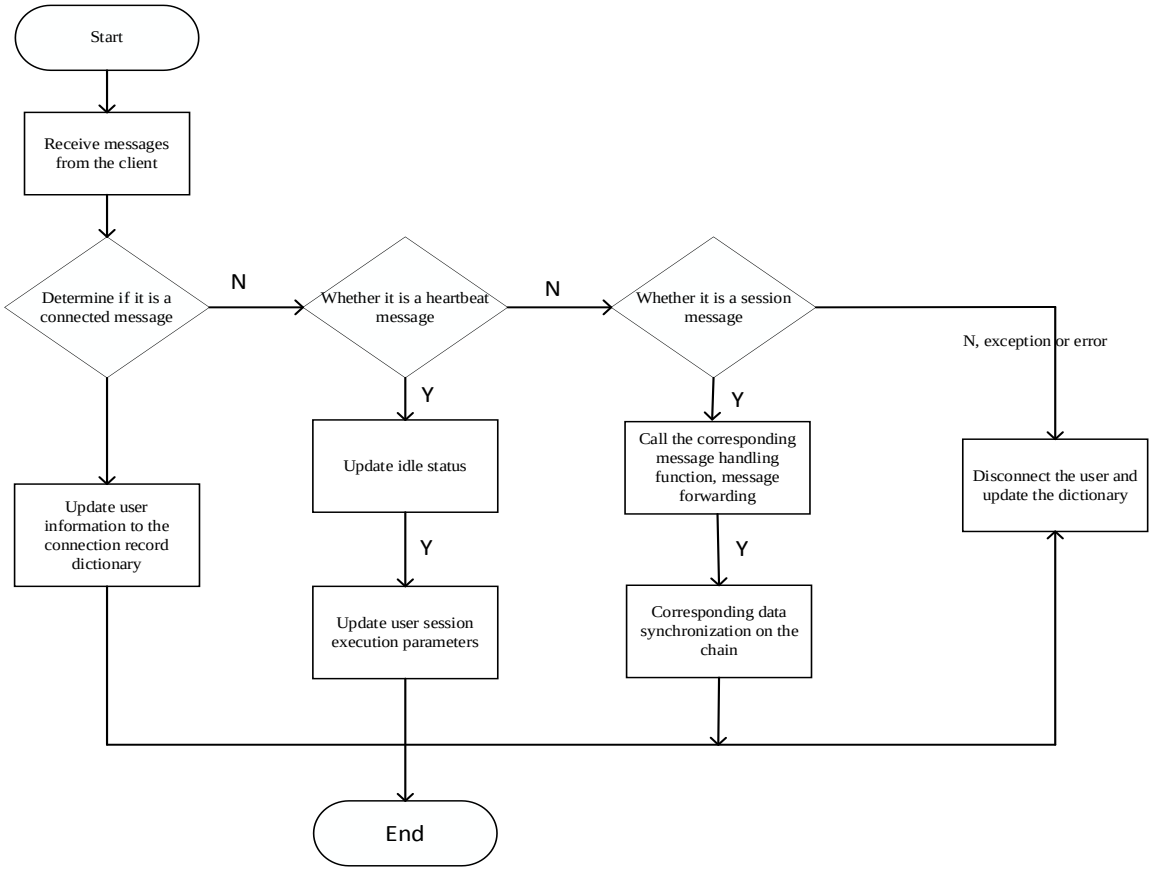

Figure 5. Alliance chain communication flow chart

2) Maintaining connections and message reception

The initialization of the federation chain bootstrap node starts the Netty service, which is shown in Fig. 6. First, the WebSocket server is initialized, i.e., the server is started to create an instance of SeverBootstrap object, after which the bound Reactor thread pool is set, i.e., the main thread pool and the worker thread pool, next, the Channel of the bound bootstrap node server is set, the ChannelPipeline is initialized, and then the Channel initializer to specify the ChanelHandler and set the business processing logic for the messages received by the channel.

In the channel initializer, the following ChannelHandler function options have been added:

Step 1: Added HTTP codecs to specify the routing format for incoming requests, allowing messages to be transmitted between users using the same protocol and format;

Step 2: Added an idle timeout check and a Handler for idle time handling to keep the connection with the user according to certain rules;

Step 3: Add a custom Handler, which is the specific business logic for receiving several types of messages.

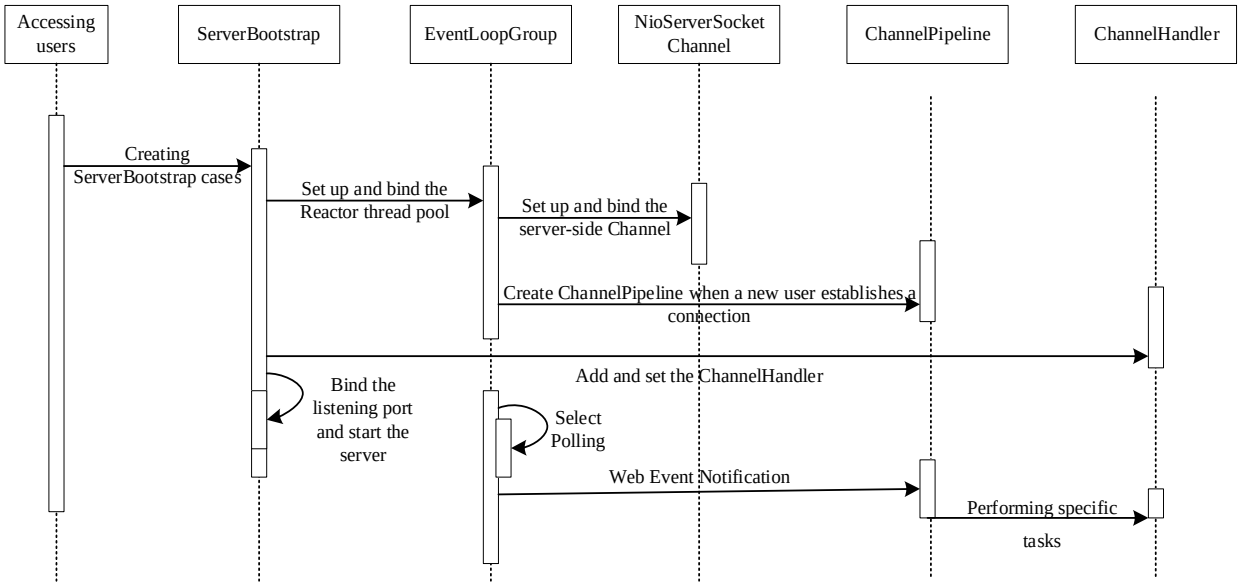

Figure 6. Netty Service Startup Timing Diagram

3) Message Parsing and Forwarding

After the bootstrap node receives the message and determines it to be a session type, it calls the relevant function to process it. The main elements of the process are.

Step 1: Message data synchronization: message data will be synchronized by a consensus algorithm to achieve forwarding between nodes and generate a unique hash, together with a summary call to the relevant function for up-linking processing.

Step 2: Adding to the message queue: placing messages that have completed consensus synchronization into the RabbitTemplate (RabbitMQ message queue interface) of the master node.

Step 3: Asynchronous forwarding of messages: take messages from the message queue, map the users corresponding to the actual keys using the corresponding derived messages, and then realize asynchronous forwarding of messages according to the processing of the received session messages.

F. Feasibility Assessment

The on-chain communication technology proposed in this system makes it possible to integrate blockchain technology into the communication field precisely by taking advantage of blockchain anonymity, tamper-proof, and traceability. Analogous to blockchain's regulation of transaction data for digital currencies, the regulation of encrypted information in communication, the assurance of tamper-proof transaction messages, and the ability to trace back absolutely true information are also a guarantee of communication security. Blockchain technology and cryptography technology have become more and more mature after years of development and have reaped many achievements in their respective fields, providing a practical basis and theoretical foundation for this system to achieve a more secure communication model, which is technically proven to be feasible.

IV. CONCLUSIONS

In this paper, we propose the KDAC algorithm, which uses blockchain technology as a communication platform and combines the extended research of ECC cryptography to reach a research design for secure communication. The research takes advantage of the huge user and natural anonymity of the blockchain network to ensure the message integrity and tamper-evident to the maximum extent, and to be able to trace back to a specific message. In addition, the key derivation scheme of one key at a time and one address at a time is researched on the basis of hierarchical key technology, and combined with the knowledge of elliptic curve cryptography, it finally realizes that the sender and receiver can communicate using new addresses and keys each time without key exchange, which further enhances the opacity of the communication itself on the basis of enhancing the opacity of the message.

REFERENCES

[1] Tounsi W, Rais H. A survey on technical threat intelligence in the age of sophisticated cyber-attacks [J]. Computers & Security, 2017: S0167404817301839.

[2] Stojmenovic I, Lin X. Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks [J]. Parallel & Distributed Systems IEEE Transactions on, 2001, 12(10):1023-1032.

[3] Freedman M J, Morris R. A peer-to-peer anonymizing network layer. MIT, 2002.

[4] Lu T-B, Fang B-X, Sun Y-Z, et al. A scalable anonymous communication protocol [J]. Computer Engineering and Applications, 2005, 41(7):4.

[5] Juha P. Provably Secure Covert Communication on Blockchain [J]. Cryptography, 2018, 2(3):18-.

[6] Song S., Peng W. BLOCCE+: An improved blockchain-based steganographic communication method [J]. Journal of Chongqing University of Technology (Natural Sciences), 2020, 34(09):238-244.

[7] Wang Xueli, Pei Dingyi. Theory and implementation of elliptic and superelliptic curve public key ciphers [M]. Science Press, 2006.

[8] Yuan Y, Ni XC, Zeng SH, Wang FY. The development status and outlook of blockchain consensus algorithm [J]. Journal of Automation, 2018, 44(11):2011-2022. doi:10.16383/j.aas.2018.c180268.

[9] Liu Yizhong, Liu Jianwei, Zhang Zongyang, Xu Tongge, Yu Hui. A review of blockchain consensus mechanism research [J]. Journal of Cryptography, 2019, 6(04):395-432. doi:10.13868/j.cnki.jcr.000311.