IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 21, 2021, accepted March 9, 2021, date of publication March 17, 2021, date of current version March 25, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3066184

The Notarial Office in E-government: A Blockchain-Based Solution

YING GAO[1], (Member, IEEE), QIAOFENG PAN[1], YANGLIANG LIU[1], HONGLIANG LIN[1], YIJIAN CHEN[1], AND QUANSI WEN[2], (Member, IEEE)

[1]School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China
[2]Jiangmen City Road Traffic Accident Social Relief Fund Management Center, Jiangmeng 529000, China

Corresponding authors: Qiaofeng Pan (cspanqiaofeng@mail.scut.edu.cn) and Quansi Wen (qwen2012@foxmail.com)

This work was supported in part by the Key-Area Research and Development Program of Guangdong Province under Grant 2020B010166001 and Grant 2019B010137001, and in part by the Natural Science Foundation of Guangdong Province under Grant 2018A030313351.

ABSTRACT The Notarial Office(NO), working on providing various essential certificates, still relies on manual handling and requires paper materials from other government departments. That brings lots of inconvenience. The Notarial Office rejects non-local paper materials for their lower credibility in the local place and then cannot provide cross-borders services. It also easily cause sensitive information leakage as copies of paper materials have been stored. In this case, a blockchain-based system is suitable to address challenges in this scenario because of its advantages (e.g, decentralized, immutability, transparency, auditability). We implemented this system on top of the Hyperledger Fabric. Moreover, we replace manual operations with smart contracts, set extra ledgers to off-load different types of transactions and provide encryption for private information when needed. In the end, we get an expected result. That is, the modification outperformed the unmodified network in experiments.

INDEX TERMS Blockchain, smart contract, e-government, cross-border services, electronic certificate, the Notarial Office.

I. INTRODUCTION

Government certificate plays an important role in the daily life of citizens in many countries. However, lack of transparency, excessive bureaucracy, and even cases of corruption, cause a decline of trust of citizens in public administration [1]. Many countries have been seeking to arrest this trend by all means. In China, the Notarial Office (NO) provides most of the government certificates to prove estate ownership, family relationship, death, etc. The establishment of a Notarial Office (NO) is to unify the certification format, reduce the number of certificate documents, improve the credibility and acceptability of certificates. The NO needs documents signed by other government departments to provide a specific certificate. Just in case, they also archive copies of these documents and leave the name of the person in charge.

The NO is essential in studying abroad, inheriting, and confirming legal person authorization. In fact, an application for studying abroad requires a birth certificate, and the inheritance requires certificates of relatives, which both signed by the Notarical Office. Moreover, it has been reported that some people lost nearly $10 million dollars due to the negligence of NO.

As a government agency, the NO has most of the shortcomings mentioned above. To some extent, lots of processes in the NO have been rendered ineffective and personal liability. Moreover, the NO easily leaks sensitive information because of archiving copies of personal data. The processes for applying for a certificate is shown in Figure 1:

1) people collect paper materials required by the NO from some local governments agencies.
2) those materials will be submitted to the NO.
3) the NO manually review paper materials and rejected those not signed by the local government to ensure the reliability of paper materials.
4) the NO provides a specific certificate if all materials are validated.

Obviously, paper materials from other cities are not completely credible, which cumbersome the cross-border services of the NO. Therefore, coordination among different

The associate editor coordinating the review of this manuscript and approving it for publication was Wenbing Zhao.

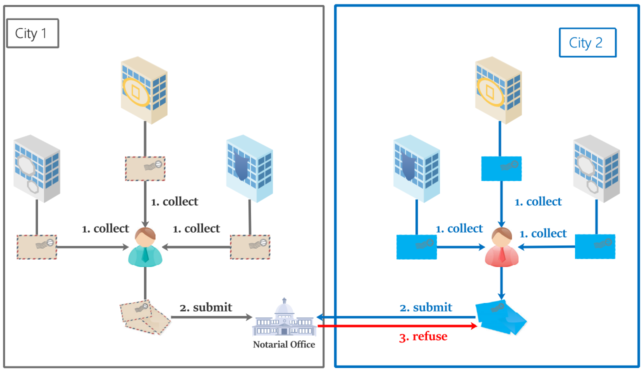

FIGURE 1. The applying procedure for certificates.

governments at different levels is limited, and it is difficult to get cross-border certificates. Besides, those consecutive procedures are not transparent, which cannot be effectively supervised and easily leading to the leakage of sensitive personal information.

The certificates provided by the Notarial Offices require manual handling at present. With the in-depth development of information and communication technology, it seems more suitable to implement services of the NO by those technologies. However, the traditional e-system usually built on centralized infrastructure, which is vulnerable and heavily controlled by third parties. That means it is prone to a single point of failure and users inside can easily alter the contents stored in the system. Besides, developing an e-service system based on centralized infrastructure can be complex and expensive, due to multiple inter-agency interactions, human labors involved, and the distances between different areas.

To overcome the challenges mentioned above, blockchain technology is a better method, compared with a centralized implementation. Blockchains are the distributed ledgers whereby participants can interact with each other in a secure, immutable way without any intermediaries. Since that the NO still relies on persons to handle materials and provide services, there is no need to consider the cost of transition between the two different systems, namely distributed system and centralized system, if the blockchain-based system is directly implemented to improve work efficiency. Besides, the way that the blockchain proves identity and sends messages makes the transmitted information between two peer nodes have high credibility, and the smart contracts can replace the manual handling with automatically process. Therefore, blockchain technology can support the cross-borders services of the NO and reduce the number of misbehaviors. That means a system building in the blockchain structure has high availability and transparency.

This article demonstrates the feasibility and effectiveness of implementing a blockchain-based system to improve the quality of NO services and provides experimental data to reference. This system, which is built on top of the Hyperledger Fabric, has two types of ledgers according to the destination of transactions, given that the distance of different cities in a large country may cause a big delay and low TPS (Transactions per second). Type one is called the local ledger and the other one is called the global ledger. The local ledger mainly processes the affairs that happened in the local city and the global ledger process the cross-border affairs. Besides, a symmetric encryption function has also been provided to protect personal sensitive information. We believe that the proposed architecture can be widely used not only for the NO services but also for other fields.

To sum up, following are the major contributions of this article:

- an analysis of a blockchain-based system for the Notarial Office is provided. We found that a blockchain-based solution is suitable for digitizing the Notary Office services. Furthermore, we proposed a better blockchain network structure to improve performance.
- we propose a decentralized system based on blockchain to meet the demands of the Notarial Office and conduct a series of experiments to investigate the performance and feasibility of this system. The recommended cross-border structure in blockchain network for the NO is implemented and the usability of the Advanced Encryption Standard(AES) encryption algorithm in this system has been tested.
- we verified the efficiency and feasibility of our proposed blockchain-based method for the Notarial Office by the tool Hyperledger Caliper. The experiment results demonstrate that the proposed structure for cross-border services outperformed the traditional blockchain network settings especially in scalability and the AES encryption can reach the same level of performance as non-encryption when used in the general size of the content of the certificate.

The remainder of this article is organized as follows. Section II presents existing related literature toward E-government. In section III, blockchain concepts and its components utilized in this article will be introduced. The system architecture is described in Section IV following by the implementation of the BC-based NO system in Section V. The experimental results are given in Section VI, while the analysis and discussion about the system are provided in Section VII. Finally, Section VIII concludes this article and outlines future work.

II. LITERATURE REVIEW

Blockchain was first introduced in [2], aiming at creating a decentralized and trustable cryptocurrency that can avoid some financial risk. It is usually associated with virtual currency bitcoin since the bitcoin project is the most successful and well-known application of blockchain [3]. In recent years, bitcoin and the underlying blockchain technology have obtained significant acceptance [4]. Nowadays, the blockchain technology has become a leading and promising technologies and implemented beyond cryptocurrency use case, especially in Internet of Thing(IoT) [5], supply chain management [6], Electronic health [7], financial application [8], crowdsourcing [9] and E-government [10].

There are currently some blockchain frameworks providing adaptable and flexible platforms to implement a variety of applications, e.g, Hyperledger Fabric, Ethereum, EOS. When referring to performance, those frameworks usually be measured in terms of throughput, latency, and scalability [11]. Authors in [12] present the performance analysis of both Ethereum and Hyperledger Fabric, indicating that the performance of Hyperledger outperforms that of Ethereum.

Blockchain can be utilized to improve government service in efficiency and effectiveness(e.g., transparency, lower costs, accurate record-keeping) [13]. It has a promising future in optimizing the business processes through secure sharing of data [14]. Authors in [15] describe two perspectives for governments in relation to the rise of BC architectures and applications. On the one hand, they adopt Blockchain technology for their own processes or services. The other perspective is to seek BC Governance to fulfill the public values and societal needs. Reference [16] depicts a Blockchain system using proof-of-concept(POC) consensus to simplify the visibility of shared data among various stakeholders as well as deploying smart contracts to facilitate decision automation in the modification of cell towers and the building. Beris *et al.* in [17] describe a concrete example in Geek. They integrated Linked Data and Distributed Ledger technologies based on BitCoin to transform the services of the Greek public sector such as Diavgeia and Nomothesia into decentralized, trusted, intelligent, and linked applications. They can reach an average throughput of 16000 decisions per day, which satisfying the regular workload of a month. In [18], the authors developed a private blockchain on Microsoft Azure Blockchain Workbench for different departments of local government and implemented smart contracts as well. They discovered that a standard virtual machine on the cloud server as a blockchain node can meet the daily needs. Nour Diallo *et al.* in [19] adopt smart contracts and Decentralized Autonomous Organization(DAO) to build an automated blockchain system to improve the efficiency and transparency of the e-government system. They demonstrate the life cycle of a government contract and indicate that most government works do not have a high requirement on latency and the performance of the blockchain system is heavily affected by the underlying blockchain infrastructure. In [20], authors show the feasibility of transforming an e-government service into blockchain-based government service under different hardware and network requirements. Moreover, they describe the clear advantages of the blockchain system in cross-border services in terms of usability and synchronization.

Using blockchain technology to offer a public notary service can also enables some activities with the public and private sectors such as residency approaches in Estonian [21]. Chenfu Xu *et al.* in [22] build an electronic certificates catalog sharing system(ECCS) based on the Hyper Fabric(v1.1) for all scenarios related to electronic certificates sharing. Their scheme designs and simulates this system based on the three-level electronic certificate architecture and the results show that it can significantly reduce the data traffic size of electronic certificates. Pengbin Han *et al.* in [23] provide a digital copyright certificate storage and trading alliance chain system based on fabric, and it gains a good performance in tamper-proof, copyright traceability, short registration time, and low cost. In [24], Dubai Economic Department(DED) build a system on the Hyperledger Fabric(v1.1) for the alignment of the business registry and unifying licensing processes and they plan to extend this implementation to include license consumers in the next phase. Reference [25] develops a blockchain certificate system on the Ethereum platform, but they only store the serial number of the certificate in the blockchain. Reference [26] build a blockchain certificate system on the Ethereum platform as well, and they mainly focus on adopting smart contract for role definition, authority management, data anti-tampering, and privacy protection. Authors in [27] design a digital education certificate prototype utilizing the permissioned framework Hyperledger Fabric(V1.4). They test the prototype by Hyperledger Caliper instrument and the significant throughputs of getting and creating transactions are respectively 1982.6 tps and 263.9 tps. To seek a higher throughput, [28] propose an educational certificate blockchain using the cooperation of peers to create blocks in place of the competition. Besides, they build a tree structure to provide an efficient query and support historical transactions query. In [29], Nguyen et al utilize the blockchain technology of Ethereum to issue immutable digital certificates, but it is needed to be continuously improved since the shortcoming of Ethereum such as scalability and operational cost.

In general, blockchain technology is convenient for electronic government services. However, researchers usually focus on building their system on the blockchain infrastructure. They seldom take governments in different areas into consideration when discussing the feasibility. Indeed, people also seek higher throughput and lower latency while the performance of a blockchain system is inevitably affected by the distances between governments and the different development degree of cities. Therefore, we propose a method that takes this factor into consideration and reaches a higher performance especially applying blockchain technology in cross-border services.

III. PRELIMINARIES

A. BLOCKCHAINS

In principle, the blockchain itself is a distributed append-only database, which maintains a list of ordered records called blocks [30]. A block contains different types of transactions, e.g., stocks, bonds, and real estate. Each block has a timestamp and cryptographic link of the previous block to prevent the content of blocks from being modified.

The shared ledger in the blockchain is jointly maintained by different nodes according to the specific consensus algorithm. According to the way that nodes join the blockchain network, the blockchain can be divided into two types, permission-less blockchain and permissioned blockchain [31]. The former is the so-called public blockchain, in which any node can join and leave the network at will, and has the qualification and opportunity to access and write to the ledger. Typical representatives include Buterin *et al.* [32], and their nodes obtain write permissions to the shared ledger through the Proof-of-Work (PoW) and Proof-of-Stake (PoS) consensus mechanisms respectively. In contrast, permissioned blockchain usually refers to consortium blockchain or private blockchain, that is, only organizations or nodes with specific permissions can join the blockchain network and perform specific functions, such as reading, accessing or writing to the shared ledger. The most popular representative is Hyperledger Fabric (HLF) [33].

Blockchain technology also presents several limitations in different sectors. It cannot completely support a high throughput scenario as a relatively low rate of transactions [34]. For example, Bitcoin takes around 10 minutes to confirm a transaction and can only achieve 7 transactions per second [35]. Furthermore, the system implementing blockchain requires more resources for security, especially for public blockchains. Each node in the blockchain network maintain the same ledger and need to validate transactions when receiving blocks from other peers.

B. RAFT CONSENSUS

Raft is a consensus algorithm for managing replicated logs and provides a better foundation for building practical systems [36]. Raft can also be applied in the Blockchain. There are several consensus algorithms for blockchain business application, e.g., proof-of-work(PoW) [2], proof-of-stake(PoS) [37], practical Byzantine fault tolerant(PBFT) [38], Paxos [39], and Raft. Among them, PoW and PoS algorithms are mainly used in the permissionless blockchain and others are common choices of permissioned blockchain. Compared with PBFT and Paxos, the Raft algorithm has high efficiency and simplicity and it has been widely adopted in the distributed systems [40]. In general, the Raft algorithm can be simply decomposed into two phase: leader election, replication, safety.

1) LEADER ELECTION

Leader election is to choose a leader when there is no leader in the cluster of nodes. A node is in one of three states: leader, follower, or candidate. There is only one leader in this cluster and the other nodes are followers, which simply respond to requests from leaders and candidates. A follower becomes a candidate when a follower receives no messages, and a candidate becomes the leader if it obtains the most votes in the cluster. Denote the vote from a follower F_i to a candidate C_j as $m_{ij} = 1$ and denote the candidate C_j have not obtained the vote from the follower F_i as $m_{ij} = 0$. We can get the leader L:

$$L = argmax_{C_j \in C^n} \sum_{F_i \in F^n} m_{ij} \tag{1}$$

$$L > \frac{1}{2}|F^n| \tag{2}$$

All nodes in the cluster of Raft are all followers initially. If a follower has not received a message that indicates one node is the leader over a period of time, it will choose a new leader as a candidate. After that, this candidate will vote for itself and vote for other nodes parallelly in the cluster until a leader is selected or the period of time is running out. Besides, Raft randomizes the time of elections to ensure a candidate can get a majority quickly.

2) REPLICATION

In general, a leader only appends a new entry to its ledger, and the followers follow the state machines of the leader. A new entry usually contains a command to operate on the state machine along with the term number. After receiving a new entry, the leader will apply the new entry to its state machine and returns the results to other nodes. Besides, a commit is successful when a majority of the nodes have replicated the new record from the leader.

However, the inconsistent still happen when the leader crashes. A follower may miss some entries when a new leader is elected. In this case, the Raft algorithm will force the follower to duplicate the ledger of the leader. That means the conflicting entries will be overwritten according to the records from the leader. To be more specifically, the overwritten entries is after the lastest entry where the two ledger agree. For determining the point of lastest entry fo each ledger, the leader maintains a *nextIndex* for each follower.

C. HYPERLEDGER FABRIC

HLF is an open-source permissioned blockchain infrastructure, which is designed to serve as the basis for developing applications or solutions with a modular architecture. It allows plug-and-play components, such as consensus mechanisms and membership services. Its modular and universal design meets a wide range of industry use cases. Compared with other blockchains, HLF has the obvious features as following.

1) MULTIPLE NODE TYPES

Nodes are communication entities of blockchain. There are two types of nodes in HLF, namely peer node and orderer node. For peer nodes, they can assume multiple roles according to the configuration of the network. For example, the Endorser peers are bound to the smart contracts to check and endorse the transaction proposal and calculate the result of the transaction execution; all the peers are Committer peers, who are responsible for receiving the transaction proposal and checking the legality again before accepting the transaction result, and writing the final result into the blockchain ledger; Anchor peers get information from the orderer nodes, save the block and update the world status of the ledger. For orderer nodes, they are responsible for sorting all the transactions sent to the network and packing the sorted transactions into blocks. Then the packed blocks are submitted to the Committer peers for processing. HLF classifies nodes according to different responsibilities, which greatly improves the efficiency of transaction processing.

2) PLUGGABLE CONSENSUS

Fabric supports the pluggable use of the consensus mechanism. So far, there are three consensus plug-ins that fabric can choose to use, namely Solo, kafka, and Raft [41]. The Raft consensus plug-in was introduced after Hyperledger Fabric 1.4.1. Compared with the existing Solo consensus and Kafka consensus, the Raft consensus is more suitable for the production environment. The reason is that Solo only has a single orderer node, and Kafka needs to rely on Zookeeper [42] cluster management to achieve node consensus. Both of these consensuses are not truly decentralized, while Raft can truly achieve decentralized distributed consensus. Raft can achieve the same efficiency as Paxos [43] consensus and produce the equivalent results as Paxos, but it is easier to understand, which makes it has better advantages in practical system deployment.

3) CHAINCODE

The smart contract in HLF is called chaincode [44], which is divided into system chaincode and user chaincode. The system chaincodes are used to realize the system-level function, including system configuration, endorsement, verification, etc., and will automatically complete the registration and deployment when the peer node starts. The user chaincodes realize the application function of the user, interacting with the ledger through the interface provided by HLF. Each chaincode is compiled into a stand-alone application running in an isolated docker container, which not only meets the various business needs of users, but also greatly reduces the security risk.

4) CHANNEL

It is essentially an isolated channel of transaction information. HLF allows participants to establish different trading channels according to specific business. Different channels achieve information isolation, and have a separate transaction ledger, which can only be accessed by participants in this channel. The emergence of the channel makes the transaction visible only to the participants in the same channel, which meets the needs of some confidential transactions and improves the flexibility of HLF business expansion.

IV. ARCHITECTURE

In this section, we elaborated on the proposed system architecture, HLF platform setup, and functionalities.

A. SYSTEM ARCHITECTURE

The inclusive idea is to convert the original manual mechanism into a blockchain-based electronic system and establish different ledgers to off-load different types of transactions according to different regions simultaneously.

In this structure, we assume that all institutions participating in the consortium have undergone strict audits as they are government agencies. Thus, those institutions, as peers in the blockchain network, have sufficient capacity to compensate for losses to others caused by improper operations. Besides, the materials required by certificates from each institution do not need to contain detailed contents. The identity authentication mechanism (e.g, MSP component in the HLF framework) in the blockchain network ensures the high credibility of materials, even there is only a simple or short description. In particular, all functions are implemented in the form of Smart Contracts deployed in a blockchain network to reduce the errors from the manual operation and improve the efficiency in all. Note that all the formats of certificates or procedures of validation in the NO services can be covered because of the Turing-completeness programming languages supporting in the HLF framework.

As depicted in Figure 2, every city has its own ledger and shares a global ledger with other cities in the proposed BC-based system. For a certificate to be used in a local city, a resident should submit the required documents and choose the kind of certificate to apply for (e.g, online, or off-line) (step 1). Next, government agencies that received requests interact with the BC platform and then propose a transaction containing corresponding materials to the local ledger (step2). The BC platform will respond to different entities after validating proposals, and then broadcast transactions (step3). As for a certificate to be used in other cities, an outlander can contact the government agencies in birthplace by any means. After that, the government in birthplace will save transactions on the global ledger. Every city included in the BC-based system can generate specific certificates after the required materials are recorded and synchronized in the global ledger (step 4). In the end, people can obtain certificates from the NO despite they are not in their birthplace.

B. WORKFLOW OF SYSTEM

Our system can be mainly divided into 6 parts: Blockchain Client and Application, Membership Service Provider(MSP), Certificate Authority(CA), the infrastructure of the blockchain node, Smart Contracts(SC), Ordering Service. At first, this system has to establish different channels to determine the peers which will be included in the distributed ledgers, which means, we can set different clusters of ordering service for various ledgers. The users in applications should register and enroll with the CA, and then receive back necessary cryptographic material. After that, they get permission from MSP to access the network. MSP is a PKI-based implementation. It is used to define the admin of an organization and allow other organizations to validate that entities have the authority to

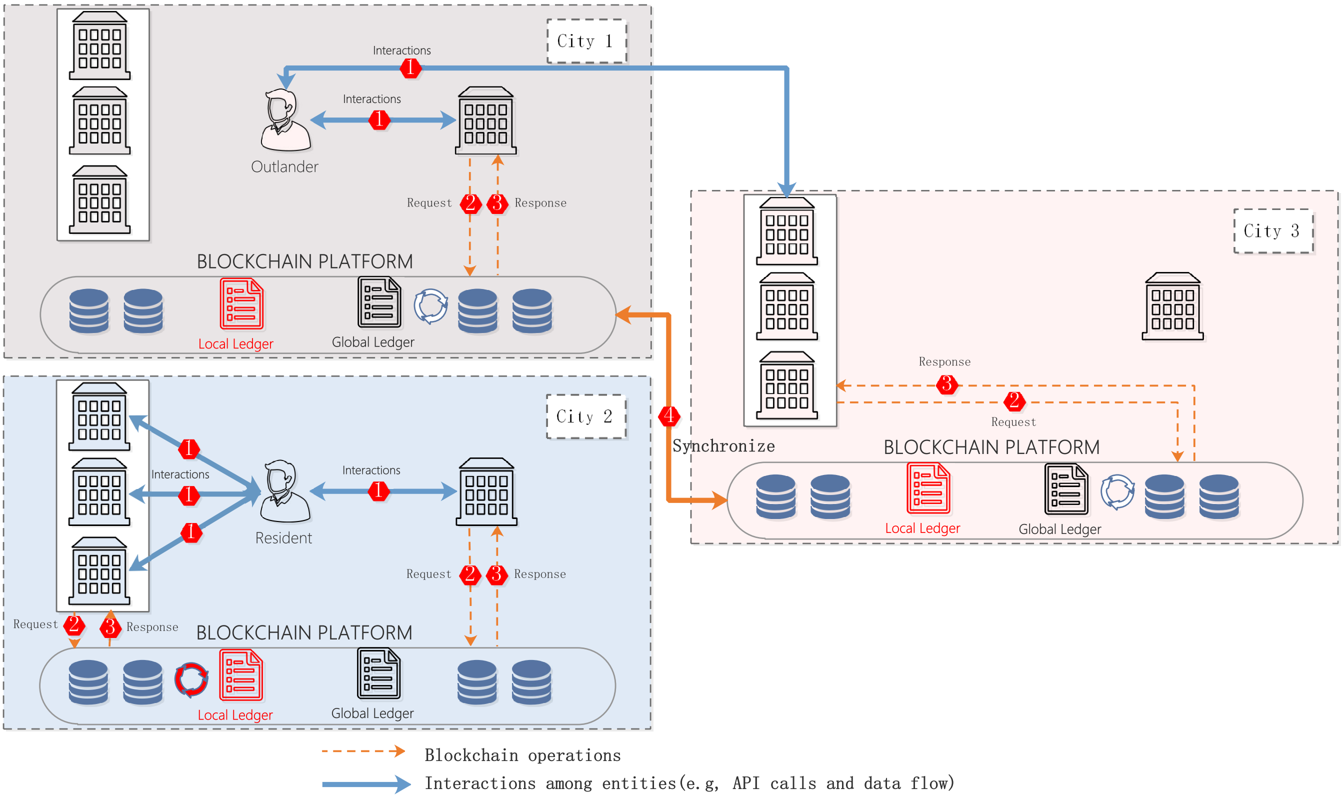

FIGURE 2. High-level system architecture of the design concept for a BC-based government certification management platform.

do what they are attempting to do. The most important thing that accounts for validating authority in MSP is the certificate issued by the CA. CA provides and dispenses X.509 certificates which are used to identify components belonging to an organization. Besides, the X.509 certificates can also be used to verify transactions to indicate that a peer endorses the transaction result. The infrastructure of the blockchain is always associated with SC. The infrastructure provides the distributed ledger service for each node, e.g, communicating with other peers, broadcasting the transactions. Furthermore, before they can contact each peer, a common set of contracts covering common terms, data, rules, concept definitions, and processes should be defined. In the end, these contracts lay out the whole model that governs all of the interactions between transacting parties. Ordering Service is also known as orderer. Instead of PoW structure, a collective of nodes are defined to order transactions into a block and distributes blocks to connected peers for validation and commit and the other nodes should provide signed responses for the validation of orderers.

As Fig 3 can see, when a user or the NO send a request by the Client, the application leveraging a supported SDK(Node, Java, Python) utilizes one of the available API to construct a transaction proposal. During this process, the operation will be permitted by the MSP component if it satisfies the policy for the node. The proposal, which is a request to invoke an SC function with certain input parameters, intends to read or update the ledger. Among the Global Peers or Local Peers, the endorsing peers defined in the Endorsement Policy will verify the proposal but do not update the ledger at this point. They also request some X.509 certificates from CA by utilizing the MSP API to validate the identity of some peers. Thereafter, those peers invoke the SC function and return a set of values along with their signatures as the signed response to the SDK. The SDK can parse the messages for applications to use. When received the responses, the application verifies the signatures and compares the proposal responses to determine whether the responses are the same. Next, the Client assembles endorsements into a transaction and broadcasts it to the ordering service. The ordering service simply receives transactions and orders them sequentially by channel, and then creates blocks containing transactions per channel. Those blocks will be delivered to all peers on the channel. The transactions included in the block are also validated by peers, and then tagged as being valid or invalid. Every peer will append the block to the corresponding chain, besides, it will notify the client application that the transaction has been appended to the chain, as well as the states of the transaction, namely validated or invalidated.

C. HLF PLATFORM SETUP

We chose the Hyperledger Fabric Blockchain framework (HLF), the most popular permissioned BC framework, as our underlying platform.

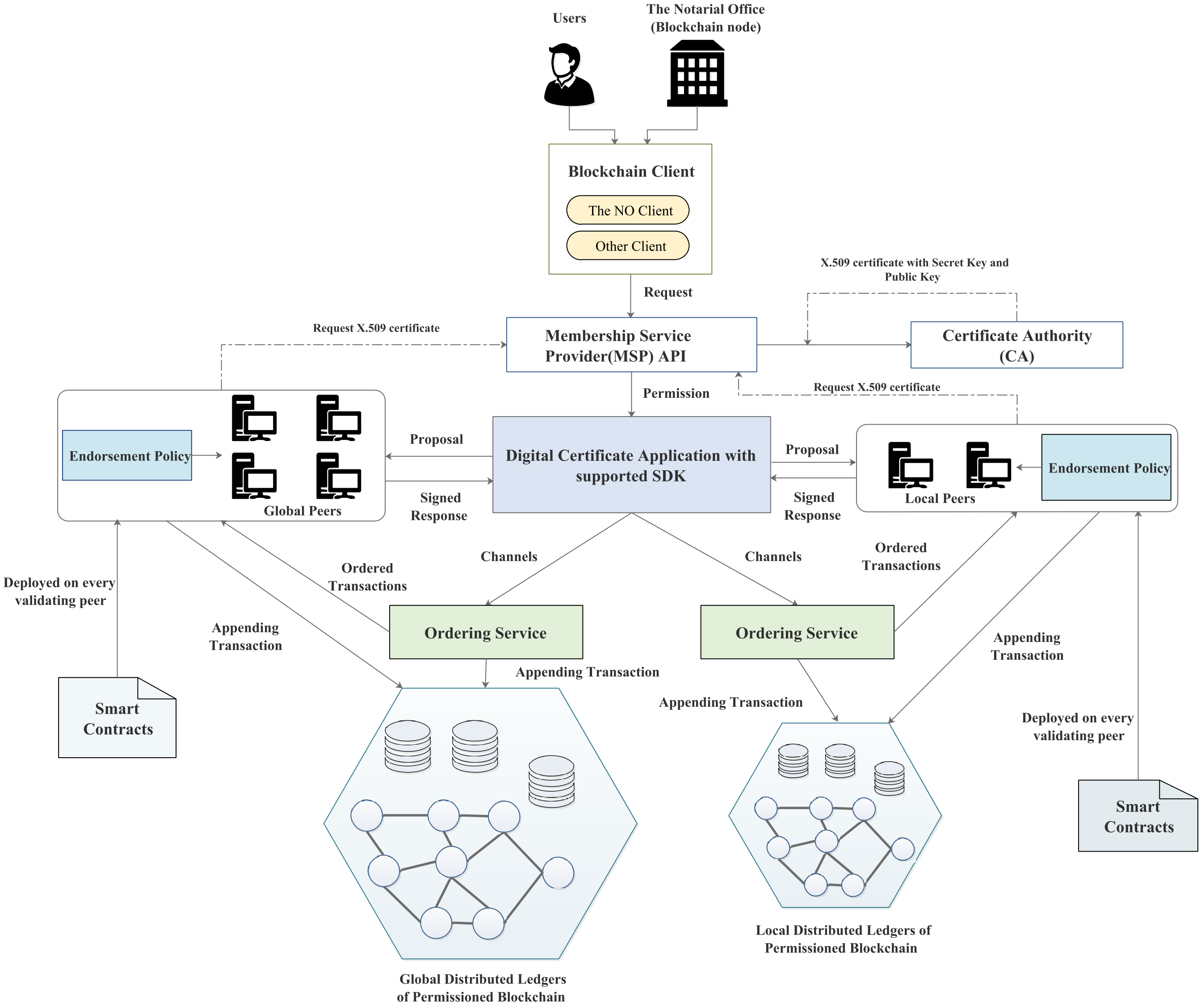
Users
The Notarial Office (Blockchain node)
Blockchain Client
The NO Client
Other Client
Request
X.509 certificate with Secret Key and Public Key
Request X.509 certificate
Membership Service Provider(MSP) API
Certificate Authority (CA)
Permission
Request X.509 certificate
Endorsement Policy
Global Peers
Proposal
Signed Response
Digital Certificate Application with supported SDK
Proposal
Signed Response
Local Peers
Endorsement Policy
Channels
Channels
Ordered Transactions
Ordered Transactions
Deployed on every validating peer
Appending Transaction
Ordering Service
Ordering Service
Appending Transaction
Appending Transaction
Appending Transaction
Deployed on every validating peer
Smart Contracts
Smart Contracts
Local Distributed Ledgers of Permissioned Blockchain
Global Distributed Ledgers of Permissioned Blockchain

FIGURE 3. The workflow of system.

Fig 4 shows the transaction flows in the proposed blockchain platform. Firstly, all government agencies in the same city belong to the same organization. The peer nodes share an identical global ledger and store a local ledger corresponding to their organization. MSP component provides a service to permit a new node to operate on a specific ledger. The CA in the MSP generates cryptography materials for the new node so that this node can communicate with other nodes in a secure, credible way. In fact, the HLF framework supports a peer node to create or share more than two ledgers, which can be adapted to subarea furthermore. Secondly, clients send transactions to peers owning the target ledger in step (1). When transactions are validated in process of simulation in step (2). those peers having the target ledger will return their digital signatures on received transactions to clients in step (3). Thirdly, the clients collect all signatures and send them to Ordering Service Nodes (OSNs) in step (4). Finally, the OSNs adopting Raft consensus review the signatures according to verify policy in step (5), and then broadcast validated transactions to other peers in step (6).

In this paper, different HLF networks are deployed, including 1, 3 or more OSNs running in Raft mode to provide the ordering service, from 4 to 35 peers, and different environments to test their throughout. All peers maintain two types of ledger (i.e., channels in HLF terminology), namely Local Ledger and Global Ledger in the experiments. Every Peer and OSN belong to those two different type channels. However, both channels have the same structure and execute the same SC(i.e., chaincode in HLF term). The only difference is the users permitted to operate them. The Local Ledger can only be operated by the peers corresponding to the organization, but the Global ledger is shared with all peers in this platform.

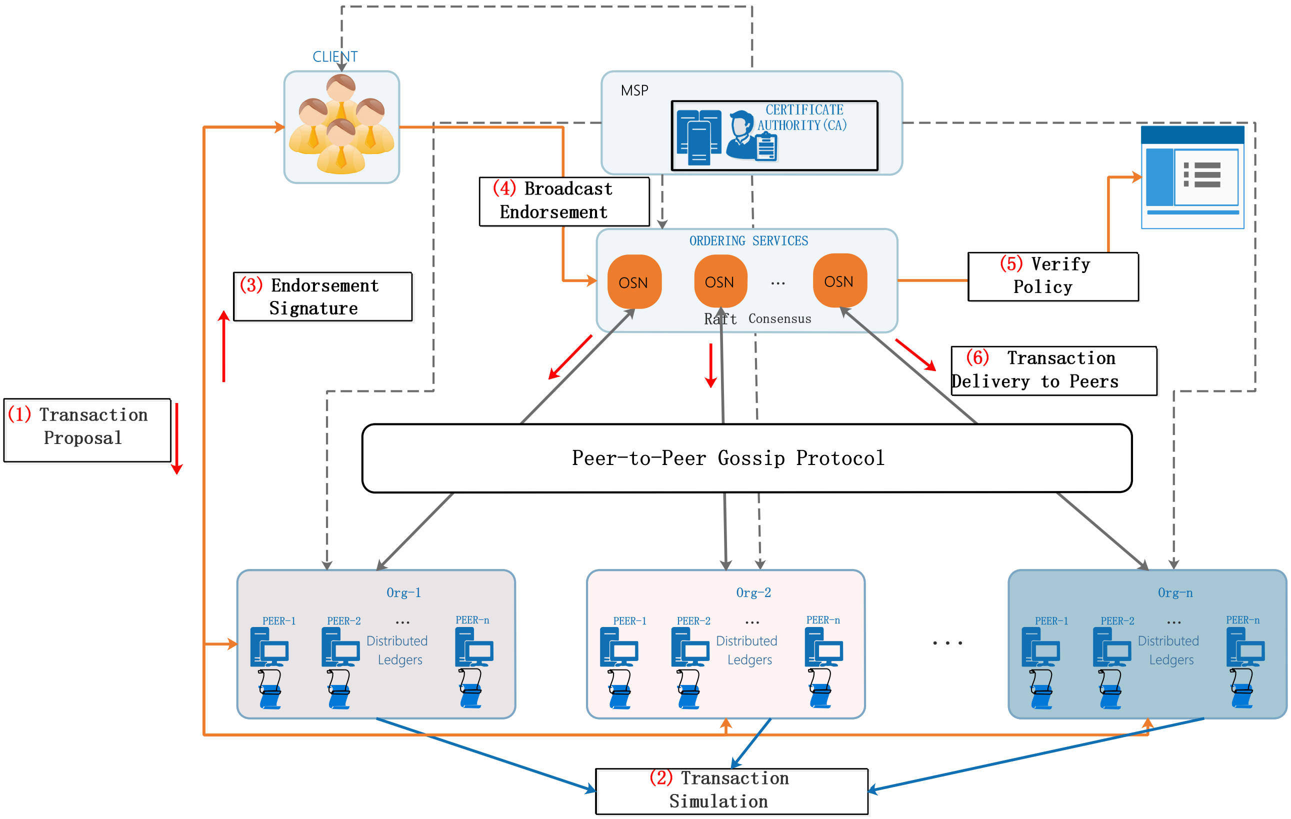

FIGURE 4. High-level system architecture and transaction flow of the HLF framework.

A client can use the Fabric Client SDK(e.g, NodeJS, Java, and Go)to interact with the HLF network, while it takes lots of time to construct a program for testing the performance of the whole system. Instead, the Hyperledger Caliper can be used to construct a transaction proposal to get the maximum load of the system. Therefore, we utilized it to simulate different situations.

D. FUNCTIONALITIES

1) MATERIAL CREATION

The government agencies write electronic documents in the form of a transaction when receiving the request for providing material to the NO. The algorithm of creating materials is given in Algorithm.1. The Blockchain platform generates a unique type_ID at first. After that, the original description from those agencies will be transformed into JSON format to more easily extract essential attributes for the NO. Next, the government agencies send both JSON messages and signatures to OSNs. In the end, the NO can get credible proof broadcasted by the OSNs in the blockchain platform.

2) CERTIFICATE CREATION

The NO creates a certificate by Smart Contract that also collects and validates materials from other government agencies. More details are shown in Algorithm.2. The users need to provide an ID number or something else to identify the resident or outlander. After that, the Smart Contract extracts information from materials according to the required items and determines the correctness of them. If everything goes smoothly, the Smart Contract will output the certificate. Otherwise, it returns errors.

3) CONTENT ENCRYPTION

To better protect privacy, the Smart Contract provides encryption functionality. The misbehavior from the operator easily causes personnel sensitive information leakage since all certificates and other relevant materials will be kept in a great number of peers. Therefore, a suitable encryption functionality is necessary. We chose the AES symmetric algorithms considering the relatively high performance. As Algorithm.1 and Algorithm.2 both show, peer nodes will encrypt the information according to the password provided by users before writing in the ledger. In this case, other peers only get the ciphertext if not providing the correct secret key and the leakage of sensitive information can be prevented to a large extent.

4) CONTENT READING

The Smart Contract can restrict the information presented to the NO or other government agencies. In many scenarios, many government agencies give more details than the NO required in the material, e.g, to prove the ability to pay off debts of someone, the government agencies used to give more detailed information about assets of the requester, but the NO

Algorithm 1 Material Creation

```
Input: $ID$, $val\_list$, $type$, $pw$
Output: $state$                         ▷ represent successful or not
  procedure createMaterial($ID$, $val\_list$, $type$, $pw$)
      $type\_ID \leftarrow$ Generate($ID$, $type$)
                                        ▷ indicate content type for user
      $exist \leftarrow$ Get_State($type\_ID$)
                                        ▷ find the material whether exists
      if $exit$ is true then
          $state \leftarrow$false
          return $state$
      end if
      $val\_JSON \leftarrow$ valueAsJSON($val\_list$)
                                        ▷ transform values into JSON fomat
      if $pw \neq 0$ then
          $success \leftarrow$ AESEncryption($val\_JSON$, $pw$)
                                        ▷ encrypt the content using $pw$
          if $success$ is false then
              $state \leftarrow$false
              return $state$
          end if
      end if
      $success \leftarrow$ Put_State($type\_ID$, $val\_JSON$)
      if $success$ is true then
          $state \leftarrow$true
          return $state$
      end if
      $state \leftarrow$false
      return $state$
  end procedure
```

Algorithm 2 Certificate Creation

```
Input: $ID$, $type\_cert$, $pw$
Output: $cert\_ID$                      ▷ the unique certificate identifier
  procedure createCertificate($ID$, $type\_cert$, $pw$)
      $check\_list \leftarrow$ returnType($type\_cert$)
            ▷ return the require types of materials for certificate
      for every $type$ in $check\_list$ do
          $exist \leftarrow$ check($type$, $ID$)
                            ▷ check the required material whether exist
          if $exist$ is false then
              $cert\_ID \leftarrow 0$
              return $cert\_ID$
          end if
      end for
      $cert\_ID \leftarrow$ Generate($ID$, $type\_cert$)
                          ▷ generate the unique number for cert_ID
      $val\_JSON \leftarrow$ Create($ID$, $type\_cert$)
                              ▷ generate the content for cert_ID
      if $password \neq 0$ then
          $success \leftarrow$ AESEncryption($val\_JSON$, $pw$)
                                              ▷ encrypt the content
          if $success$ is false then
              $cert\_ID \leftarrow$0
              return $cert\_ID$
          end if
      end if
      $success \leftarrow$ Put_State($cert\_ID$, $val\_JSON$)
      if $success$ is false then
          $cert\_ID \leftarrow$0
          return $cert\_ID$
      end if
      return $cert\_ID$
  end procedure
```

just requires specific part of it. Therefore, when peer nodes execute the operation of reading, they only provide limited information by the Smart Contract.

V. CASE STUDY

In this part, we will introduce our framework in detail in the issuing birth certificate scenario, which is part of the daily routine of the Notarial Office. In the traditional way, a man who wants to get the birth certificate from the NO has to provide a paper certificate about the relationship between him and his parents, a paper certificate signed by the hospital to prove his birthdate, the marriage certificate about his parents, and the proof of residence provided by the local government agencies. Notice that most of the organizations that provide the materials are in the local city that the man lives in, it hard to confirm the validity of the materials for the cross-border services in the NO. Actually, there are some other issues mentioned previously, e.g, the limited coordination between different government agencies, low transparent of operation on the personal privacy, low efficiency caused by manually procedures, single point of failure, but they can be improved if we apply the framework proposed in this paper.

A. SUBMIT REQUESTS

Suppose that a man named Bob wishes his birth certificate can be validated by organizations in another city. Firstly, he should pay a higher price than the local ledger for submitting his requests on the global ledger through the blockchain client. Those requests are relevant to the required materials for generating the birth certificate, and the materials will be in the format of an electronic certificate in our framework instead of papers. The client will transform those requests into transaction proposals and send them to different peers corresponding to different organizations. Moreover, an organization can own lots of peers and the client can set the endorsement policy containing some specific organizations to endorse proposals.

B. VERIFY IDENTIDY

The proposals belong to Bob will be verified by peers, and need to get endorsements from them. We leverage an asymmetric cryptography named The Ellptic Curve Digital Signature Algorithm (ECDSA) to assure the security of the communications. Asymmetric cryptography is also known as public-key cryptography. It is more convenient and provides robust authentication as the privacy remains intact. In our framework, each entity has its certificate that contains the public key and stored in the MSP component. Furthermore, every entity owns a role assigned by the MSP component. The detailed procedures for secure communications are shown as follows:

- $Setup(1^\lambda) \rightarrow sk$: A new entity can choose a random number as its private key by this function. Each entity should register in the Fabric CA with the public key generated from the private key. After that, a certificate related to this new user will be stored in the CA server, and this certificate will be sent for validating the identity of signing some transaction.
- $KeyGen(sk) \rightarrow pk$: The method to generate the public key is based on the equation 3. This equation is an elliptic curve on a graph, p denotes a prime number. Nonetheless, there is a limited range of values since we are dealing with integers. Besides, the curve parameters a and b often are picked from some pre-made and standardized curves that are known to be secure and efficient.

$$y^2 = (x^3 + a \times x + b)\ mod\ p \tag{3}$$

$$P = sk \times G \tag{4}$$

  Equation 4 is the definition of generating the public key, that is, multiplying sk and point G which is selected randomly in the curve. The multiplication in this curve can be regarded as the addition of the point P to itself sk times, which is also a trap door function verified in [45]. That means, even the P and G are known, there still is no way to determine the exact value of sk. In the end, we can obtain the point P and choose the x coordinate of P as the public key.
- $Sign(sk, m) \rightarrow ct$: To make sure the integrity of the information and avoid tampering, the entities involved in communication have to sign the message sent by themselves. An entity that wants to sign a message should generate a random value k, and calculate R in the same way as the public key by the equation 4.

$$z = func_{SHA-1}(m) \tag{5}$$

$$S = k^{-1}(z + sk \times R) \quad mod \quad p \tag{6}$$

  Simultaneously, it also needs to make an SHA-1 hash of the message like equation 5 to get a huge number called z. Next, the value of S can be calculated by equation 6, and the pair value (R, S) together is the ECDSA signature ct.
- $Verify(pk, ct)$: Verifying the signature from another entity is simple, the receiver just needs to calculate equation 7 to calculate the P'. If the x coordinate of the point P' is equal to R, the signature is valid, else it is truthless.

$$P' = S^{-1} \times z \times G + S^{-1} \times R \times pk \tag{7}$$

C. GET ENDORSEMENTS

Although the signature is validated, Bob's proposal still needs to be verified that (1) it is well-formed, (2) there is no same proposal submitted already, (3) the client that send this proposal has the authority to operate this ledger. If all the conditions mentioned above satisfied, Bob's proposal will be input into the smart contract as the arguments, and then the SC will return the results which is the main content of the endorsement for the peer. However, there is no updating to this ledger in this phase. Finally, the endorsement of a peer for the proposal of Bob will be in the form of results obtained from SC along with the signature of the peer.

D. COMMIT CERTIFICATE

The client verifies the signatures and compares responses to determine if those responses are the same. If the request from Bob is only querying the ledger, the client would only inspect the response. Otherwise, the client will check whether the endorsement policy has been fulfilled before submitting Bob's transaction to the ordering services. In the case of submitting a transaction, the ordering service only receives transactions of different ledgers and creates blocks of transactions by chronological order.

At this time, the block containing the information related to the materials for generating the birth certificate will be broadcasted to all the peers on the global ledger. Each peer appends the block to the chain of the ledger. Eventually, Bob can interact with the client to send a request, which calls the SC to generate the birth certificate based on the information stored in the global ledger in the way we have described.

VI. IMPLEMENTATION

The proposed platform in this paper will process a large number of transactions not only in the local city but also across borders among lots of areas. Therefore, throughput, latency, and scalability are the most important metric to evaluate performance. Furthermore, given that this system is applied to the government, a strict mechanism to enroll users is recommended to use in this system. Thus, a permissioned blockchain framework is optimal to implement this system. Actually, the Hyperledger Fabric framework has demonstrated promising advancements in both performance and scalability. In a word, we have chosen the HLF platform to implement the proposed system. The following subsections show details of our implementation with system configurations.

A. IMPLEMENTATION ENVIRONMENT

We have implemented various experiments on servers equipped with 4 Intel CPU@2.2GHz processor and 8G RAM. The blockchain network that we use has been deployed in Ubuntu 16.04(64-bit), using RAFT-based Fabric in version 2.1.0 [46]. Fabric is an extensible blockchain system for running distributed applications, separates transaction execution from consensus and enables policy-based endorsement [47].

The government agencies that belong to identical organization occurring in the experiment uses a server to run the order and peer nodes, while the other trusted institutions utilized other servers.

B. HYPERLEDGER CALIPER

For better performance evaluation, we choose a tool developed by the Hyperledger Foundation named Caliper [48]. Caliper is a blockchain performance benchmark framework that can test various blockchain network frameworks with customing use cases, including Hyperledger Besu, Ethereum, Hyperledger Fabric, and FISCO BCOS. In short, Caliper generates a workload against a specific framework under test and then continuously monitors its responses [49]. Finally, the Caliper will give a report based on the responses observed.

The Caliper supports HLF systems in version 2.x at present. It needs a network and benchmark file in the form of YAML configuration, which both have been written at genesis, to test an existing blockchain network. We selected Throughput and Latency, which are provided by reports of Caliper, as our measurement parameters considering the efficiency of the metric.

VII. EXPERIMENTAL RESULTS

A. PERFORMANCE COMPARISON

Fig 5 and Fig 6 illustrate the difference in performance between Global Ledger and Local Ledger under an increasing number of transactions, from 10 to 1 million. Both of them are tested under the HLF network including 4 peer nodes, 2 organizations, and 1 OSN. Besides, only 1 peer is used to send transactions to get a more precise result.

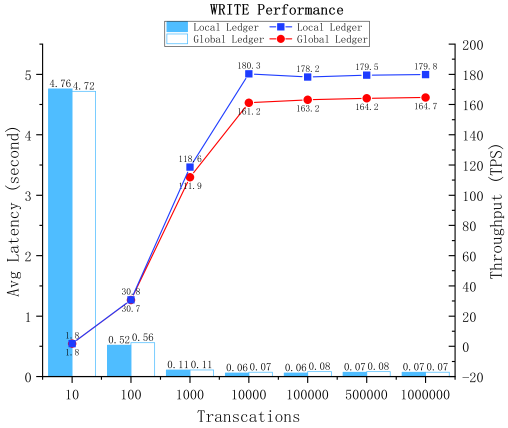

FIGURE 5. Performance of WRITE in different type of ledgers.

The given graph outlines that throughput in the Local Ledger is higher than the Global Ledger. Even their TPSs are similar at the task of 10 or 100 transactions, writing TPS in the Local Ledger becomes faster than writing in the Global Ledger at the task of 1000 and 1 million transactions. Especially, this trend keeps stable when transactions above 10000. On the other hand, although both average latencies keep a downward tendency, there is no obvious difference between values in the end and more likely to reach the bottom point. As for the reading performance of ledgers, all values obtained from the two types of ledgers are similar.

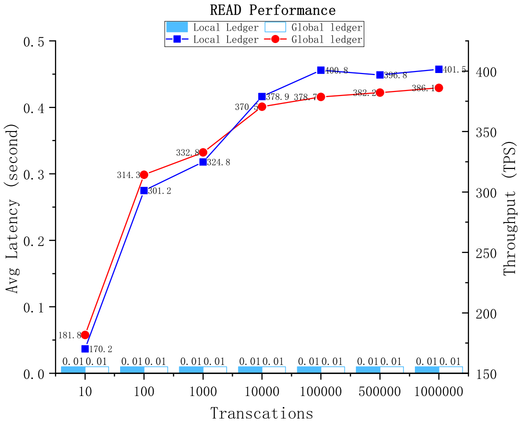

FIGURE 6. Performance of READ in different type of ledgers.

We can easily conclude that the TPS of the local ledger is greater than the global ledger under the same network conditions. It can be easily understood since the local ledger needs fewer peers to endorse than the Global ledger. In fact, TPS will sharply decrease if all peers from different cities operate a ledger simultaneously. Therefore, transferring transactions to different ledgers according to the type can improve the efficiency of the BC network, i.e., combining the local ledger and the global ledger is an alternative way to improve the performance bottleneck of blockchain applications

B. SCALABILITY

To further explore, we gradually increased the number of peer nodes, but do not change the number of organizations and order peers, and then we used Caliper to submit transactions in different organizations at the same time. We have 3 order peers in this experiment and denote a city as an organization. Moreover, the fixed number of organizations also means the number of the local ledgers is fixed. In this experiment, 20% of transactions are submitted to the Global ledger which has 3 order peers, and 80% of transactions are sent to a unique ledger that has 1 order peer and belongs to the corresponding organization. The sum of transactions is also up to 1 million. We also write all the data into the Global Ledger and regard it as our comparative experiment.

Fig 7 depicts the reading and writing performance of the proposed HLF network. The writing operation, which occupies most of the operation time of the blockchain network, can rise to nearly 1.8 times, and the reading rate can also be improved. In addition, every time adding 3 peers to this network, the proposed HLF network only reduces almost 8% performance, while the traditional network, namely all nodes only sharing one ledger, will reduce about 12% performance.

Besides, we establish another experiment that keeps the number of peer in an organization but increase the organizations and order peers. The organizations also represent

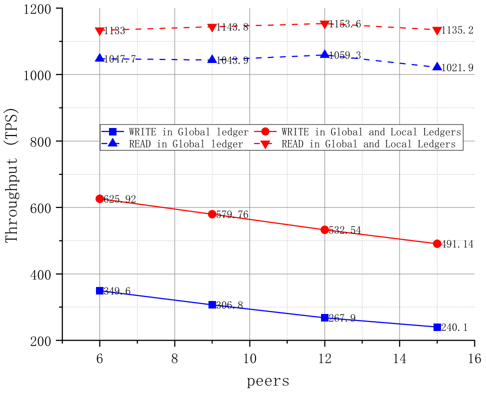

FIGURE 7. Scalability of the proposed HLF network when peers are added.

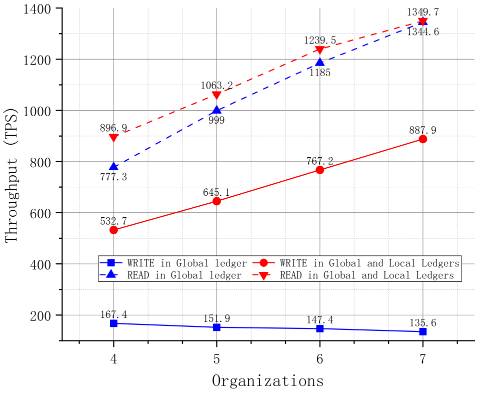

FIGURE 8. Scalability of the proposed HLF network when organizations are added.

different cities, which all own their local ledger. Each organization owns 5 peers and 1 order peer. We submit the transactions in the same way as the previous experiment. Finally, we get the results shown in figure 8.

Fig 8 demonstrates that the reading performances between the global ledger and the framework proposed in this paper are similar, while the writing performance is very different. The throughout of reading is increasing with the organization enlarging. It makes sense since reading the ledger does not increase the burden to order peers and the more organizations means the more server. Moreover, submitting transactions at different organizations simultaneously only occupies the resources of the corresponding server.

As for the writing performance, our proposed framework outperforms the only global ledger in fig 8. The TPS of our method is increasing, but the TPS of the global ledger is decreasing. Given the total amount of transactions up to 1 million, the added organization that owns 5 peers and 1 order peer will proportionally off-load the work of the whole blockchain network. That means there are more ledgers that process the transactions at the same time, which leads to higher performance.

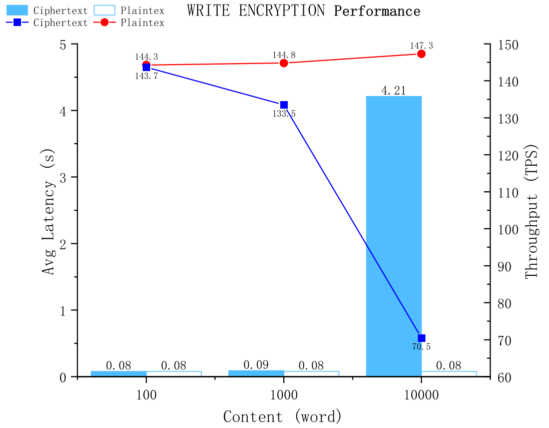

FIGURE 9. Performance of WRITE in different type of ledgers.

In conclusion, the scalability of the proposed blockchain networks in this paper is better than that of traditional blockchain networks. In this blockchain network, a peer node affords all throughputs of two ledgers. However, fewer nodes to endorse transactions in the Local Ledger lead to a lower decrease in performance when adding peer nodes. Furthermore, more transactions are off-loaded on the newly created ledgers that cause the increment of TPS.

C. ENCRYPTION PERFORMANCE

To prevent the leakage of sensitive information, we choose the AES algorithm to encrypt transactions considering its wide use and high efficiency. Although the efficiency of the various algorithm affected by the difference parameter, AES algorithm gain the most performance in many use cases such as time-consuming, response time, the request executed per second, and battery power consumption [50]. Therefore, the AES algorithm is suitable to keep a high frequency of transactions and encrypting the certificates simultaneously. Besides, there is no need to induce some other components to implement this algorithm, which means not increasing the complexity of our framework. From the users' perspective, they only need to remember a series of words or numbers instead of storing the information related to encryption on a device.

In this experiment, we tested the performance of this encryption function by constantly adding words in the HLF framework with 1 organization and 5 peers. Fig 9,10 describes the encryption effects on transactions. The increasing of characters makes the TPS drop, both in writing and reading, but for the latency, it increases significantly around 10,000 transactions and not obvious at the beginning when writing. Hence, we can easily get the conclusion that the rate of decreasing per 100 words is relatively low.

Actually, certificates with more than 10,000 words are seldom used in real life. As the performance the graphs depict, we can use the AES algorithm to encrypt the most transaction and obtain a similar performance compared with plaintext.

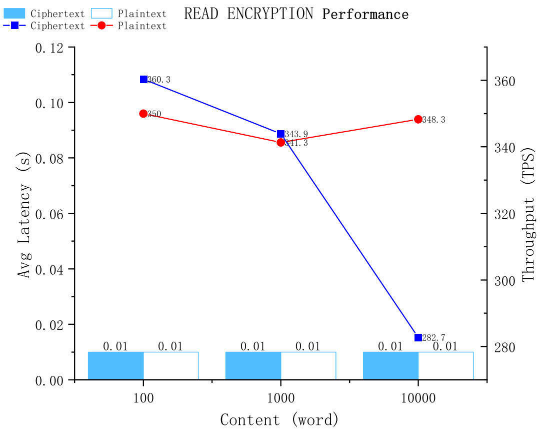

FIGURE 10. **Performance of READ in different type of ledgers.**

VIII. SECURITY ANALYSIS AND DISCUSSION

The advanced BC frameworks(HLF v2.1) can prevent most security threats. For better analysis, some specific issues details show as follow:

1) Falsify Material: Given that peers in the permissioned blockchain have high credibility, this happens relatively rarely. Even if this happens, a clear accountability mechanism can minimize the impact of generating counterfeit materials by internal adversaries. When a peer creates fake materials, the record will be stored in every peer owning the updated ledger. Therefore, it has to compensate for losses caused by relevant certification.
2) Wrong Operation: Manual operations bring great discretion to some extent, but the discretion can be restricted through automatic execution in the form of smart contracts. Besides, all updates on the ledger will be recorded in the blockchain, which provides trusted audit and trace when disputes happen.
3) Access Control: The access control pattern in this system relies on the MSP component provided by HLF. The Membership Service Provider(MSP) identifies specific privileges of an actor on a node or channel. A user registered with a Fabric CA will be associated with the role of admin, peer, client, orderer.

IX. CONCLUSION

This paper proposes an electronic certificates sharing system based on consortium blockchain to address the challenges of cross-border government services, especially in terms of auditability, efficiency, and privacy. We evaluate the performance through a prototype implementation. In this work, we analyzed the demands of the Notarial Office, which provides certificates for residents in China, and discover that a blockchain-based solution can address most challenges occurring in the NO. Besides, we have modified the widely used structure of the blockchain network to improve the performance. The applied modification is to classify all transactions into the local transactions and global transactions and then off-load to different ledgers. This method achieves a sound performance in the experiments. We can also consume that the actual performance will higher as there are the distance metric and development degree of cities to be taken into account. Lastly, we provide security analysis on various aspects, showing the advantages of our proposed solution.

In this paper, we just establish the same type of ledgers in our experiments, including a global ledger and different local ledgers representing different cities, and our future work is communication on inter-blockchain. It can help to improve the efficiency and saving the space of storing transactions, e.g, the records on the local ledger have no need to stored on the global ledger and can be leveraged to generate the certificates as materials in our scenario. However, we have to take the reliability of information across the different blockchains into consideration, but for the government routines, each organization has a high level of trust. That means we can directly use the records from the local ledger which belongs to one city as the extended information for the global ledger in some scenarios.

REFERENCES

[1] T. PERSSON, C. F. PARKER, and S. WIDMALM, "Social trust, impartial administration and public confidence in EU crisis management institutions," *Public Admin.*, vol. 95, no. 1, pp. 97–114, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/padm.12295

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

[3] T.-T. Kuo, H. Zavaleta Rojas, and L. Ohno-Machado, "Comparison of blockchain platforms: A systematic review and healthcare examples," *J. Amer. Med. Inform. Assoc.*, vol. 26, no. 5, pp. 462–478, May 2019, doi: 10.1093/jamia/ocy185.

[4] Y. K. Tomov, "Bitcoin: Evolution of blockchain technology," in *Proc. IEEE XXVIII Int. Sci. Conf. Electron. (ET)*, Sep. 2019, pp. 1–4.

[5] T. A. Alghamdi, I. Ali, N. Javaid, and M. Shafiq, "Secure service provisioning scheme for lightweight IoT devices with a fair payment system and an incentive mechanism based on blockchain," *IEEE Access*, vol. 8, pp. 1048–1061, 2020.

[6] Y. P. Tsang, K. L. Choy, C. H. Wu, G. T. S. Ho, and H. Y. Lam, "Blockchain-driven IoT for food traceability with an integrated consensus mechanism," *IEEE Access*, vol. 7, pp. 129000–129017, 2019.

[7] A. Shahnaz, U. Qamar, and A. Khalid, "Using blockchain for electronic health records," *IEEE Access*, vol. 7, pp. 147782–147795, 2019.

[8] Y. Yan, C. Wei, X. Guo, X. Lu, X. Zheng, Q. Liu, C. Zhou, X. Song, B. Zhao, H. Zhang, and G. Jiang, "Confidentiality support over financial grade consortium blockchain," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, New York, NY, USA, Jun. 2020, p. 2227, doi: 10.1145/3318464.3386127.

[9] S. Zhu, Z. Cai, H. Hu, Y. Li, and W. Li, "ZkCrowd: A hybrid blockchain-based crowdsourcing platform," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4196–4205, Jun. 2020.

[10] S. Ølnes and A. Jansen, "Blockchain technology as s support infrastructure in e-government," in *Electronic Government*, M. Janssen, K. Axelsson, O. Glassey, B. Klievink, R. Krimmer, I. Lindgren, P. Parycek, H. J. Scholl, and D. Trutnev, Eds. Cham, Switzerland: Springer, 2017, pp. 215–227.

[11] M. Kuzlu, M. Pipattanasomporn, L. Gurses, and S. Rahman, "Performance analysis of a hyperledger fabric blockchain framework: Throughput, latency and scalability," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 536–540.

[12] S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, "Performance analysis of private blockchain platforms in varying workloads," in *Proc. 26th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2017, pp. 1–6.

[13] D. Yermack, "Corporate governance and blockchains," *Rev. Finance*, vol. 21, no. 1, pp. 7–31, Mar. 2017.

[14] A. Kaur, A. Nayyar, and P. Singh, "Blockchain: A path to the future," *Cryptocurrencies Blockchain Technol. Appl.*, pp. 25–42, May 2020, doi: 10.1002/9781119621201.ch2.

[15] S. Ølnes, J. Ubacht, and M. Janssen, ''Blockchain in government: Benefits and implications of distributed ledger technology for information sharing,'' *Government Inf. Quart.*, vol. 34, no. 3, pp. 355–364, 2017.

[16] H. Treiblmaier and C. Sillaber, *A Case Study of Blockchain-Induced Digital Transformation in the Public Sector*. Cham, Switzerland: Springer, 2020, pp. 227–244, doi: 10.1007/978-3-030-44337-5_11.

[17] T. Beris, I. Angelidis, I. Chalkidis, C. Nikolaou, C. Papaloukas, P. Soursos, and M. Koubarakis, ''Towards a decentralized, trusted, intelligent and linked public sector: A report from the greek trenches,'' in *Proc. Companion Proc. World Wide Web Conf.*, May 2019, pp. 840–849.

[18] V. A. Bharadi, P. P. Ghag, S. R. Chavan, S. S. Gawas, and A. Kazi, ''Integrating blockchain with local public service system,'' in *Proc. ICBCT*, D. Patel, S. Nandi, B. Mishra, D. Shah, C. N. Modi, K. Shah, and R. S. Bansode, Eds. Singapore: Springer, 2020, pp. 93–103.

[19] N. Diallo, W. Shi, L. Xu, Z. Gao, L. Chen, Y. Lu, N. Shah, L. Carranco, T.-C. Le, A. B. Surez, and G. Turner, ''EGov-DAO: A better government using blockchain based decentralized autonomous organization,'' in *Proc. Int. Conf. eDemocracy eGovernment (ICEDEG)*, Apr. 2018, pp. 166–171.

[20] D. Geneiatakis, Y. Soupionis, G. Steri, I. Kounelis, R. Neisse, and I. Nai-Fovino, ''Blockchain performance analysis for supporting cross-border E-Government services,'' *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1310–1322, Nov. 2020.

[21] C. Sullivan and E. Burger, ''E-residency and blockchain,'' *Comput. Law Secur. Rev.*, vol. 33, no. 4, pp. 470–481, Aug. 2017.

[22] C. Xu, H. Yang, Q. Yu, and Z. Li, ''Trusted and flexible electronic certificate catalog sharing system based on consortium blockchain,'' in *Proc. IEEE 5th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2019, pp. 1237–1242.

[23] P. Han, A. Sui, T. Jiang, and C. Gu, ''Copyright certificate storage and trading system based on blockchain,'' in *Proc. IEEE Int. Conf. Adv. Electr. Eng. Comput. Appl. (AEECA)*, Aug. 2020, pp. 611–615.

[24] S. N. Khan, M. Shael, and M. Majdalawieh, ''Blockchain technology as a support infrastructure in E-Government evolution at dubai economic department,'' in *Proc. Int. Electron. Commun. Conf.*, New York, NY, USA, Jul. 2019, p. 124, doi: 10.1145/3343147.3343164.

[25] J.-C. Cheng, N.-Y. Lee, C. Chi, and Y.-H. Chen, ''Blockchain and smart contract for digital certificate,'' in *Proc. IEEE Int. Conf. Appl. Syst. Invention (ICASI)*, Apr. 2018, pp. 1046–1051.

[26] R. Xie, Y. Wang, M. Tan, W. Zhu, Z. Yang, J. Wu, and G. Jeon, ''Ethereum-Blockchain-Based technology of decentralized smart contract certificate system,'' *IEEE Internet Things Mag.*, vol. 3, no. 2, pp. 44–50, Jun. 2020.

[27] H. Cheng, J. Lu, Z. Xiang, and B. Song, ''A permissioned blockchain-based platform for education certificate verification,'' in *Blockchain and Trustworthy Systems*, Z. Zheng, H.-N. Dai, X. Fu, and B. Chen, Eds. Singapore: Springer, 2020, pp. 456–471.

[28] Y. Xu, S. Zhao, L. Kong, Y. Zheng, S. Zhang, and Q. Li, ''ECBC: A high performance educational certificate blockchain with efficient query,'' in *Theoretical Aspects of Computing (ICTAC)*, D. V. Hung and D. Kapur, Eds. Cham, Switzerland: Springer, 2017, pp. 288–304.

[29] D.-H. Nguyen, D.-N. Nguyen-Duc, N. Huynh-Tuong, and H.-A. Pham, ''CVSS: A blockchainized certificate verifying support system,'' in *Proc. 9th Int. Symp. Inf. Commun. Technol. (SoICT)*, New York, NY, USA, 2018, p. 436, doi: 10.1145/3287921.3287968.

[30] F. Casino, T. K. Dasaklis, and C. Patsakis, ''A systematic literature review of blockchain-based applications: Current status, classification and open issues,'' *Telematics Informat.*, vol. 36, pp. 55–81, Mar. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0736585318306324

[31] A. R. Rajput, Q. Li, M. Taleby Ahvanooey, and I. Masood, ''EACMS: Emergency access control management system for personal health record based on blockchain,'' *IEEE Access*, vol. 7, pp. 84304–84317, 2019.

[32] V. Buterin. (2014). *A Next-Generation Smart Contract and Decentralized Application Platform*. [online] Available: https://cryptorating.eu/whitepapers/Ethereum/Ethereum_white_paper.pdf

[33] (2018). *Hyperledger Fabric: Hyperledger*. [Online]. Available: https://www.hyperledger.org/projects/fabric

[34] C. Fan, S. Ghaemi, H. Khazaei, and P. Musilek, ''Performance evaluation of blockchain systems: A systematic survey,'' *IEEE Access*, vol. 8, pp. 126927–126950, 2020.

[35] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. Gün Sirer, D. Song, and R. Wattenhofer, ''On scaling decentralized blockchains,'' in *Financial Cryptography and Data Security*, J. Clark, S. Meiklejohn, P. Y. Ryan, D. Wallach, M. Brenner, and K. Rohloff, Eds. Berlin, Germany: Springer, 2016, pp. 106–125.

[36] D. Ongaro and J. Ousterhout, ''In search of an understandable consensus algorithm,'' in *Proc. USENIX Annu. Tech. Conf. (USENIX ATC)*, 2014, pp. 305–319.

[37] S. King and S. Nadal, ''Ppcoin: Peer-to-peer crypto-currency with proof-of-stake,'' *Self-Published Paper, August*, vol. 19, p. 1, Aug. 2012.

[38] M. Castro and B. Liskov, ''Practical Byzantine fault tolerance,'' in *Proc. OSDI*, vol. 99, 1999, pp. 173–186.

[39] L. Lamport, ''The part-time parliament,'' in *Concurrency: Works Leslie Lamport*. New York, NY, USA: Association for Computing Machinery, 2019, pp. 277–317.

[40] D. Huang, X. Ma, and S. Zhang, ''Performance analysis of the raft consensus algorithm for private blockchains,'' *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 1, pp. 172–181, Jan. 2020.

[41] D. Ongaro and J. Ousterhout, ''In search of an understandable consensus algorithm (extended version),'' *Retrieved July*, vol. 20, p. 2018, 2016.

[42] P. Hunt, M. Konar, F. P. Junqueira, and B. Reed, ''ZooKeeper: Wait-free coordination for Internet-scale systems,'' in *Proc. USENIX Annu. Tech. Conf.*, 2010, vol. 8, no. 9, pp. 1–14.

[43] L. Lamport, ''Paxos made simple,'' *ACM SIGACT News*, vol. 32, no. 4, pp. 18–25, 2001.

[44] E. Androulaki, A. De Caro, M. Neugschwandtner, and A. Sorniotti, ''Endorsement in hyperledger fabric,'' in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 510–519.

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

[46] *A Blockchain Platform for the Enterprise*. Accessed: Dec. 1, 2020. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/release-2.1/

[47] E. Androulaki, A. Barger, V. Bortnikov, and C. Cachin, ''Hyperledger fabric: A distributed operating system for permissioned blockchains,'' in *Proc. 13th EuroSys Conf.*, Apr. 2018, pp. 1–15, doi: 10.1145/3190508.3190538.

[48] *Measuring Blockchain Performance with Hyperledger Caliper*. Accessed: Dec. 1, 2020. [Online]. Available: https://www.hyperledger.org/blog/2018/03/19/measuring-blockchain-performance-with-hyperledger-caliper

[49] *Hyperledger Caliper Architecture*. Accessed: Dec. 1, 2020. [Online]. Available: https://hyperledger.github.io/caliper/v0.4.2/architecture/

[50] M. B. Yassein, S. Aljawarneh, E. Qawasmeh, W. Mardini, and Y. Khamayseh, ''Comprehensive study of symmetric key and asymmetric key encryption algorithms,'' in *Proc. Int. Conf. Eng. Technol. (ICET)*, Aug. 2017, pp. 1–7.

YING GAO (Member, IEEE) received the bachelor's and master's degrees from Central South University, China, in 1997 and 2000, respectively, and the Ph.D. degree from the South China University of Technology, China, in 2006. She is currently a Professor with the School of Computer Science and Engineering, South China University of Technology. She has published more than 30 papers in international journals and conferences. Her current research interests include service-oriented computing technology, software architecture, blockchain, and network security.

QIAOFENG PAN received the B.S. degree from East China Jiaotong University, China, in 2018. He is currently pursuing the M.S. degree in computer science and engineering from the South China University of Technology. His current research interest includes blockchain and its application.

YANGLIANG LIU received the B.S. degree from the South China University of Technology, China, in 2018, where he is currently pursuing the M.S. degree in computer science and engineering. His current research interest includes blockchain and its application.

HONGLIANG LIN received the B.S. degree from the South China University of Technology, China, in 2018, where he is currently pursuing the M.S. degree in computer science and engineering. His current research interest includes blockchain and its application.

YIJIAN CHEN received the B.S. degree from the South China University of Technology, China, in 2018, where he is currently pursuing the M.S. degree in computer science and engineering. His current research interests include blockchain and proxy re-encryption.

QUANSI WEN (Member, IEEE) received the bachelor's and master's degrees from RMIT University, in 2011 and 2013, respectively, and the Ph.D. degree from the South China University of Technology, in 2020. She is currently a Researcher with the Jiangmen City Road Traffic Accident Social Relief Fund Management Center. Her current research interests include data analysis, blockchain technology, access control, and network security.