mathematics

MDPI

Article

A Hybrid Encryption Scheme for Quantum Secure Video Conferencing Combined with Blockchain

Dexin Zhu [1,2], Jun Zheng [1], Hu Zhou [2], Jianan Wu [2], Nianfeng Li [2] and Lijun Song [3,*]

[1] School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100811, China
[2] College of Computer Science and Technology, Changchun University, Changchun 130022, China
[3] Jilin Engineering Laboratory for Quantum Information Technology, Jilin Engineering Normal University, Changchun 130052, China
* Correspondence: ccdxslj@126.com or songlj@jlenu.edu.cn

Abstract: Traditional video conference systems depend largely on computational complexity to ensure system security, but with the development of high-performance computers, the existing encryption system will be seriously threatened. To solve this problem, a hybrid encryption scheme for quantum secure video conferencing combined with blockchain is proposed in this study. In the system solution architecture, first, the quantum key distribution network is embedded in the classic network; then, the "classical + quantum" hybrid encryption scheme is designed according to the secret level required for the video conference content. Besides, the real-time monitoring module of the quantum key distribution network is designed to ensure that users can check the running state of the network at any time. Meeting minutes can be shared by combining with blockchain. In order to quickly query meeting minutes, a cache-efficient query method based on B+ tree is proposed. The experimental results show that compared with the traditional video conference system, the quantum secure video conference system sufficiently integrates the technical advantages of the quantum key distribution to resist the security threats such as channel eavesdropping and high-performance computational attacks while ensuring the stable operation of the classic system, thus providing a video conference system with a higher security level. Meanwhile, the query time cost of blockchain with different lengths is tested, and the query efficiency of the proposed method is 3.15-times higher than the original query efficiency of blockchain.

Keywords: video conference systems; quantum key; blockchain; hybrid encryption

MSC: 68U10; 68U35; 81P45; 94A08

check for updates

Citation: Zhu, D.; Zheng, J.; Zhou, H.; Wu, J.; Li, N.; Song, L. A Hybrid Encryption Scheme for Quantum Secure Video Conferencing Combined with Blockchain. *Mathematics* **2022**, *10*, 3037. https://doi.org/10.3390/math10173037

Academic Editors: Weihua Xu, Jinhai Li and Xibei Yang

Received: 17 July 2022
Accepted: 18 August 2022
Published: 23 August 2022

1. Introduction

Information exchange and communication are tremendously facilitated by the development of the Internet. Under the general background of promoting a new era of government affairs and business applications that are convenient, efficient, and environmentally friendly, video conference systems are widely applied in various fields, such as government, military, enterprise, education, medical, and personal fields. From an enterprise perspective, the rational utilization of video conference systems can facilitate information transfer and shorten decision cycles and execution cycles, thereby reducing time and space costs. Meanwhile, it can also improve the overall operational efficiency of the enterprise, thus accomplishing eco-friendly administration. With the continuous improvement of information technology, Internet infrastructure is being constructed all over the world, and the update of video technology is quickly changing. To meet the overall demands of terminal users for high-quality video streaming, researchers began to improve key technologies such as compression algorithms, transmission delay, and the error rate. Wen Tao et al. [1] applied VPN technology to the construction of a cross-regional technology video conference network, and an enterprise video conference dynamic resource reservation algorithm with a variable bandwidth was designed by Cheng Haixiu et al. [2]. By analyzing the resource allocation of the video conference system according to different priority levels, the QoS control framework was designed with the OpenDayLight lithium-based controller [3].

In video conferences, military intelligence, business secrets, classified information, personal privacy, etc., are often involved in conference contents. Therefore, people are paying more and more attention to the security of video data and the instantaneity of transmission. According to the pseudo-randomness of the chaotic sequence cipher, the chaotic cipher generation algorithm was improved by using a two-dimensional time-varying discrete space–time system, and then, the video frame was encrypted by XOR with the new pseudo-random key stream, which improved the security and real-time performance of network multimedia video files [4]. Zhu Yanping [5] applied wavelet transform technology to video scrambling encryption and, at the same time, used chaotic keying technology to increase the amount of keys and improve the parallelism of algorithms, so as to improve the security and real-time performance of video encryption. Di Xiaoqiang et al. [6] used the chaotic sequences generated by a quantum cellular neural network to initialize the chaotic map of logistics, generated the chaotic sequence, and then, repeated bitwise operations to obtain enough keys to solve the problem of large video coding delay after the chaotic video encryption algorithm. Pseudo-random sequences were generated based on the spatiotemporal lattices, which were used for the encryption and decryption key of the image or video file [7]. According to two chaotic maps, a secure pseudo-random number generator was designed, and a binary sequence was generated by this pseudo-random number generator [8]. Reference [9] proposed an alternative video encryption algorithm, in which the key is generated pseudo-randomly by the ChaCha algorithm. The H.264/A VC encoding algorithm encodes the video into multiple slices; the key semantic elements in the slice can be selectively encrypted; the key is generated by a pseudo-random generator and updated in real-time [10]. However, in the existing scheme, although the video security problem can be solved well, the key is pseudo-randomly generated according to the traditional key negotiation process, and the security mainly depends on some number theoretic problems with computational complexity, such as the discrete logarithm problem, integer factorization problem, and elliptic curve discrete logarithm problem. With the development and implementation of high-performance computers, especially quantum computing technology, the existing encryption system that guarantees security based on computational complexity will pose a serious threat [11].

In order to deal with the security threat brought by quantum computing, a quantum secure communication scheme based on the quantum key distribution (QKD) can be adopted [12]. Quantum communication uses a single photon as the information carrier, and the single photon is inseparable. An eavesdropper cannot obtain the key information by stealing half a photon and measuring its state. The eavesdropper can measure the state of a single photon after intercepting it, but the Heisenberg uncertainty principle in quantum mechanics detects the eavesdropper's measurement of the photon, thereby verifying the security of the key established by the two communicating parties. The eavesdropper can also steal the information by copying the quantum state of the single photon, but the no-cloning theorem guarantees that an unknown quantum state cannot be exactly replicated. The QKD method realizes the generation of a key with theoretical absolute randomness and does not require a third party to transmit the key. Therefore, QKD does not depend on the complexity of computing to ensure communication security, so that the security of quantum cryptosystems will not be threatened by the continuous improvement of computing power and the mathematical level, thus ensuring the security of the encrypted information of the quantum cryptography system.

This paper proposes a hybrid encryption scheme for quantum secure video conferencing combined with blockchain, aiming at solving the security problem of data information in a classical video conference system. Firstly, the QKD network based on time phase coding technology running on commercial optical fiber is effectively combined with the classical video conference system network. The QKD network generates theoretically absolutely secure quantum keys and uses quantum keys to encrypt audio and video data information. Then, due to the low key rate of the QKD network, it cannot satisfy the one-time pad encryption of all audio and video data. A hybrid encryption scheme is proposed, that is the main site selects classical key encryption and quantum key encryption according to the security level of video conference content. Secondly, the QKD network runs in the actual commercial optical fiber, and the key distribution is easily affected by external factors. Therefore, a real-time monitoring function is designed to display the network operation status on the interface of the video conference system to ensure the stable operation of the system. Finally, with the advantages of blockchain technology, such as tamper-proofness, decentralization, and traceability, the meeting minutes are encrypted and uploaded to the blockchain (alliance chain). A cache-efficient query method based on B+ tree is designed to facilitate viewing of the meeting minutes. The blockchain information is uploaded to the cache layer, and the keyword index in the block is used as the index address. The leaf nodes in the B+ tree store the index address of meeting minutes and the ciphertext. To solve the conflict of index addresses, the leaf nodes store the ciphertext in an adaptive space. The layers above the leaf nodes are used as indexes.

Compared with the classical video conferencing system, the main contributions of this paper include the following four aspects:

(1) A networking scheme for seamlessly connecting QKD equipment to the video conferencing system is established to realize the effective combination of the quantum key and classical applications.

(2) According to the characteristic that the quantum key generation rate is lower than the quantum key consumption rate, a hybrid encryption scheme is designed. Based on the security level of the video conference data, two sets of encryption algorithms of "one-time pad + AES" quantum key encryption and "AES" classical key encryption are adopted.

(3) Since the QKD is affected by factors such as the channel, environment, and eavesdropping, the real-time monitoring function of the QKD network is designed to detect the key generation, so as to ensure the continuous and stable operation of the video conference system.

(4) Using blockchain technology to share meeting minutes. After the meeting, miners write the encrypted meeting minutes into the blockchain. A cache-efficient query method based on B+ tree is designed, and an adaptive space method is used to solve the problem of index address conflict. On the basis of not revealing the plaintext of meeting minutes, previous meeting information can be searched, decrypted, and viewed.

The other parts of this paper are arranged as follows. Section 2 explains the basic knowledge of the Quantum Key Distribution, Blockchain and B+ tree. Section 3 describes the system model. Section 4 gives the specific design of the scheme. Section 5 analyzes the security and stability of the model. Section 6 provides the system operation analysis. Section 7 summarizes the full text. Appendix A describes two algorithms.

2. Preliminaries

2.1. Notations' Definition

All the notations used in this paper are summarized in Table 1.

Table 1. Notations used in our scheme.

Notations	Definitions
F	Quantum key file
ck	Classic binary key
qk_{BA}	Bob and Alice's symmetric binary quantum key
l_i	Row i of the database
QT	Quantum key table
QT_B^l	Row l in Table B
M_i	Group i plaintext with a length of 1024 Byte
C_i	Group i ciphertext with a length of 1024 Byte
$En_{opt}(qk, M)$	Encryption operation, where opt is the encryption algorithm
$De_{opt}(qk, C)$	Decryption operation, where opt is the decryption algorithm
ID_{AES}	Quantum key ID when starting AES encryption

2.2. Quantum Key Distribution

The quantum key distribution (QKD) can enable both parties of the communication to generate and share a random and secure key to encrypt and decrypt the communication message based on quantum mechanical properties. QKD protocols mainly comprise the single-photon-based QKD protocol, the continuous variable-based QKD protocol, and the quantum-entanglement-based QKD protocol, wherein the BB84 protocol [13] is a more common protocol used in current research. The BB84 protocol, also known as the four-state protocol, was proposed by C.H. Bennett and G. Brassard in 1984. The implementation of this protocol requires two channels: one is a classic channel and the other is a quantum channel for transmitting the quantum state carrying random numbers. In the protocol, the quantum state corresponds to the classical binary code, wherein the 0° or 45° phase corresponds to the classical bit 0 and the 90° or $-45°$ phase corresponds to the classical bit 1. Use the $|\psi_{00}\rangle$ and 0° phase quantum state of the corresponding photon, the $|\psi_{10}\rangle$ and 90° phase quantum state of the corresponding photon, the $|\psi_{01}\rangle$ and 45° phase quantum state of the corresponding photon, and the $|\psi_{11}\rangle$ and $-45°$ phase quantum state of the corresponding photon. The basic principle is that Alice sends the single photon of horizontally vertical and positive and negative quantum states to Bob, and Bob measures the quantum state. After receiving all the information sent by Alice, Bob performs the base loss comparison with Alice through the classic channel; if the base loss is the same, the original quantum key is obtained. Since Eve cannot observe the photon phase quantum state without changing the state, once it is eavesdropped, the bit error rate will be significantly improved. Therefore, the key distribution security can be guaranteed in theory, thus providing a safe channel that is difficult to crack for information transmission. In addition, it can effectively deal with security threats such as channel eavesdropping and high-performance computational attacks, therefore greatly improving the encrypted communication security and ensuring the network transmission security.

At present, the construction of actual quantum networks such as quantum satellite networks, quantum metropolitan area networks, and quantum trunk networks [14], the participation of Alibaba Cloud, mobile, and telecom operators, and the integration of quantum keys and classical applications have involved many fields. QKD was used for secure data transmission in network microgrids [15]. A satellite-based QKD network was created to realize the quantum key distribution between two legitimate users and the satellite for secure communication [16]. References [17–19] introduced the combination scheme of the quantum key and power business and gave the security protection architecture of a new urban power business. The integration of quantum keys with classical applications has been involved in several fields, but in the existing schemes, most of them consider the scheme of using quantum keys and never consider the monitoring of the operation status of quantum key distribution networks. In video image encryption, Reference [20] first proposed an image encryption algorithm based on quantum cryptography. Reference [21] used a six-state scheme to generate quantum keys, performed a XOR operation with the image data, and permuted the image data to realize the encryption of images. Reference [22] obtained the quantum key according to the BB84 protocol, then fused the quantum key with the logistic chaotic sequence to generate the final key stream and permuted the image pixel values by the heteroskedastic operation to realize the image encryption. The literature [21,22] used the basic protocol of the quantum key distribution to generate quantum keys and implemented image encryption experimentally, but did not consider the relationship between the key consumption of the one-time pad algorithm and the key generation rate based on the BB84 protocol in the actual quantum key distribution network and the influence of other related factors on image encryption. References [23–26] used the quantum random walk technique to generate quantum keys and then encrypted the image information. Reference [27] fused quantum keys to chaotic sequences by nonlinear operations and used hybrid keys to encrypt images. This scheme fuses random keys with pseudo-random keys, which improves the randomness and consistency of pseudo-random keys, but reduces the randomness of quantum keys. Reference [28] proposed an image encryption scheme based on discrete-time alternating quantum walk (AQW) and the advanced encryption standard (AES), which used quantum walk technology to provide theoretically safe keys for the AES algorithm. Although References [23–26,28] used quantum random walk technique to generate quantum keys and provided a scheme for encrypting images, the quantum keys are not theoretically completely random, but have pseudo-randomness, which is different from the quantum keys generated through the quantum key distribution technique. At the same time, the generation of quantum keys is real-time, and the use of a single encryption scheme will result in the waste or tightness of quantum keys. In general, there are some image encryption schemes that introduce quantum keys, and each has its own differences, but such schemes are still relatively few.

Reference [29] introduced the quantum blockchain [30], proposed a new quantum blockchain scheme based on quantum entanglement, and analyzed the security of the transaction from the keys and quantum coins. Reference [31] used quantum keys to encrypt bidders' private data and submit them to the blockchain and designed a quantum-based auction system on the blockchain. These schemes give the combination of quantum communication technology and blockchain; for the security proof, however, no specific experimental verification and analysis have been designed. In this paper, QKD technology is introduced into the existing video conference system scheme, making full use of the physical properties of quantum keys and combining with existing schemes to effectively strengthen the system security.

2.3. Blockchain

Blockchain consists of multiple nodes, each of which is independent and has the same status. Each node keeps certain information and relies on the consensus mechanism to ensure the consistency of stored data, which can be understood as a distributed ledger technology (DLT), and DLT is jointly maintained by these nodes. The characteristics of blockchain are that data are difficult to tamper with and forge and have traceability. The blockchain records the information of all transactions; the process is efficient and transparent, and the data are highly secure. According to the degree of openness of the nodes, it can be divided into three categories: public chain, alliance chain, and private chain.

The public chain is completely developed externally, and users can directly participate anonymously and can access the network and blockchain without registration or authorization. The scope of application of the private chain is limited to private organizations, and the read and write privileges and participation in the blockchain are set according to the rules of private organizations. The alliance chain refers to a blockchain in which several institutions jointly participate in bookkeeping, that is alliance members reach a consensus through mutual trust of multiple centers. The data of the alliance chain only allows the member nodes in the system to read, write, and send transactions and record transaction data together.

Compared with the public chain, the alliance chain has more advantages in high availability, high performance, programmability, and privacy protection. It is considered to be a "partially decentralized" or "multi-center" blockchain. The number of consortium chain nodes is limited, and most nodes only need to reach a consensus to trade, while the transaction speed is fast, so in real scenarios, the consortium chain is easier to use. In the alliance chain platform, Hyperledger is a classic case. Hyperledger is a collaborative open-source project of the Linux Foundation in 2015, aiming to promote cross-industry blockchain technology. It is a cooperative project of global cross-industry leaders and has become a global technology alliance in the field of blockchain. By creating enterprise-level, open-source distributed classification frameworks and code bases, it assists organizations to expand and build industry-specific applications, platforms, and hardware systems to support their respective transaction businesses.

At present, blockchain has typical applications in all walks of life. Reference [32] designed a 2019-nCoV vaccine supply platform based on a blockchain platform. According to the tamper-proofness feature of the blockchain, it was used to verify identity, vaccine registration, storage, and side effect reporting. Reference [33] applied blockchain technology in the management of college students' transcripts. Reference [34] presented a land certificate management scheme based on blockchain. Xu Wenyu et al. [35] proposed a privacy protection scheme for electronic health records based on blockchain and homomorphic encryption, which solved the problem of electronic health record interaction. Using the combination of blockchain smart contracts and homomorphic encryption technology, the function of automatic claims' settlement was realized without revealing any sensitive user information. Yang Xiaodong et al. [36] proposed an electronic evidence sharing scheme for the Internet of Vehicles based on signcryption and blockchain, using signcryption and proxy re-encryption technology to encrypt the electronic evidence of the Internet of Vehicles, storing evidence reports on the blockchain; the evidence ciphertext was stored in the cloud server. Tan Haibo et al. [37] proposed a blockchain-based archive data protection and sharing method. This method stores archive data through a private Inter Planetary File System (IPFS) cluster and uses smart contract technology to realize the sharing and acquisition of archive data. Through the combination of the alliance chain and the public chain, the protection, verification, and recovery of archive data were realized.

2.4. B+ Tree Structure

B+ tree is a balanced multiway search tree, which is widely used in file systems. Its characteristics are that it can keep the data stable and orderly and the insertion operation has a relatively stable logarithmic time complexity. An M-order B+ tree node has the following characteristics:

(1) Each node has at most m subtrees.
(2) A node with n subtrees contains n keywords.
(3) All leaf nodes contain the information of all keywords and pointers to records containing these keywords, and the leaf nodes themselves are linked in ascending order according to the size of the keywords.
(4) All non-terminal nodes can be regarded as the index part, and the node only contains the largest (or smallest) keyword in its subtree.

In data retrieval, because B+ tree has the characteristics of strong range query and sorting ability, as well as stable efficiency, this scheme adopts 3-order B+ tree as the index structure of the data buffer. The leaf node information structure is shown in Figure 1. Each keyword index corresponds to the quantum key ID and the ciphertext information of the meeting minutes. The B+ tree structure is shown in Figure 2. In order to solve the conflict of the same keyword index value, this scheme stores the node information of the same keyword index value in an adaptive space, where it is required.

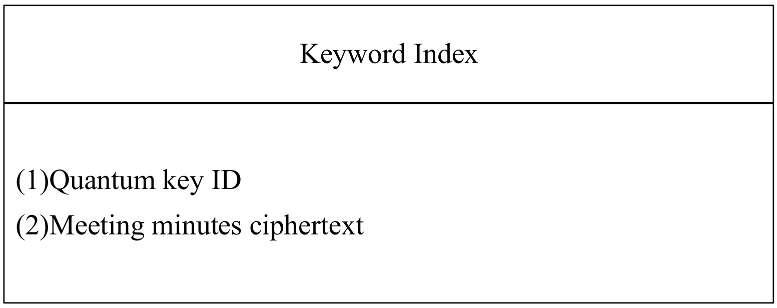

Figure 1. Leaf node structure.

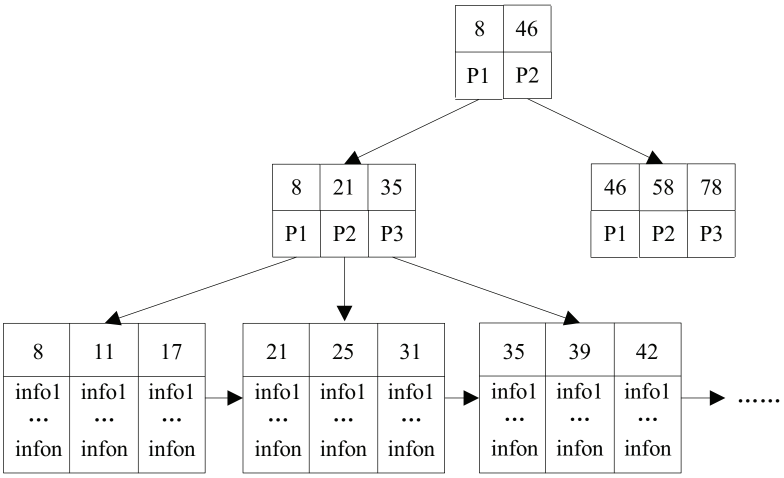

Figure 2. B+ tree structure.

To query the node information with the keyword index value *key* = 25, the following retrieval rules are required:

(1) $8 \leq key \leq 46$, select the P1 pointer.
(2) $21 \leq key \leq 35$, select the P2 pointer.
(3) In the leaf node set pointed to by pointer P2, find the node with $key = 25$ through traversal.

3. Problem Formulation

In this section, we define the problem from three aspects: system model, threat model, and security requirements.

3.1. System Model

The system solution architecture of the quantum secure video conference system combined with blockchain is shown in Figure 3. The architecture comprises users (main site and sub-site), the video conference system server (VCSS), quantum key management equipment (QKME), QKD equipment, classical channels, and quantum channels, wherein, the role of each component is as follows:

(1) Users: store video conferencing software, application platform for meeting minutes' recording, and local quantum key database. Among them, the video conferencing software includes a quantum-key-based "one-time pad + AES" encryption and decryption module, a classical key-based AES encryption and decryption module, and a QKD real-time monitoring module. Users of the main site can switch between quantum encryption and classical encryption, and it include necessary hardware for video conferences such as cameras, microphones, and monitors. The meeting minutes application platform uses the quantum key to encrypt the meeting minutes and obtains the key index value of the meeting minutes, together with the key ID used as a transaction, which is written into the blockchain by the miners. In order to facilitate users to query

the minutes multiple times, the transaction information on the blockchain is written into the application platform cache.

(2) VCSS: VCSS creates multiple tables in the database to store the symmetric quantum keys generated by each QKD device.

(3) QKME: QKME stores QKD real-time monitoring software and quantum key management software (QKMS). The QKD real-time monitoring software checks the key generation status of the QKD equipment and sends a timeout heartbeat response to the user when there is a problem with the key generation. QKMS is used to store and manage quantum keys generated by QKD devices.

(4) QKD equipment: It adopts the quantum key distribution equipment based on time phase encoding, which is mature and widely used. The generation of quantum keys and the transmission of quantum states are based on quantum channels.

(5) Blockchain: It stores the cipher text information of the video conference text minutes and has the query function.

(6) Classical channel: used to transmit classical data.

(7) Quantum channel: used to transmit quantum states and generate keys.

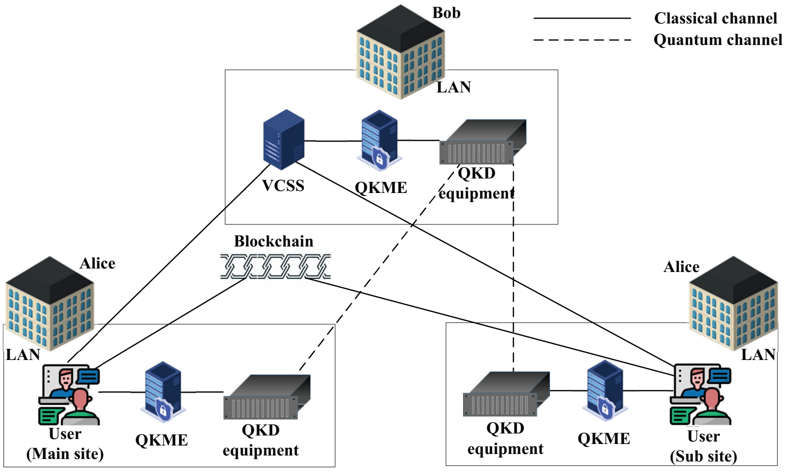

Figure 3. Scheme model.

In the model of the video conference system, we set up a standard star quantum communication network, which needs to deploy QKD devices to realize the secure communication with each other. The specific structure is as follows:

(1) Quantum key storage: The QKD equipment completes key agreement through the quantum channel and stores the Bob–Alice symmetric key in the QKME in the form of a file. Bob's QKME creates multiple key files according to the number of Alice's and stores the symmetric key matching Alice. Alice's QKME only stores the local key. Alice's user and Bob's server, respectively, read the key from the key file and store the key in the local database.

(2) Hybrid encryption scheme: The user obtains the key from the database, and according, to the security level of the video conference data, the main site user chooses to use the key scheme of "one-time pad + AES" or AES to encrypt audio and video data. The encrypted data are transmitted to the server of the video conference system through the classical channel. After the server decrypts and encrypts them, they are sent to the other user. The user uses the local key to decrypt and display the audio and video data. The running process is shown in Figure 4.

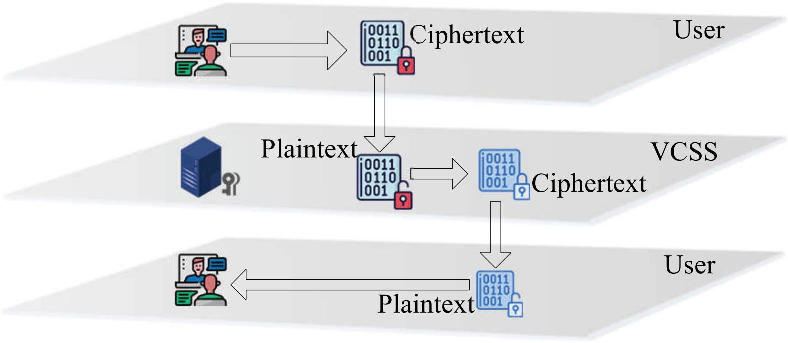

Figure 4. Data transmission between users.

(3) The real-time monitoring of QKD is shown in Figure 5. The QKME monitors the key file in real-time to check whether the file size is updated. Time interval $T_{Heartbeat} = T_{read} + T_{send} + T_{other}$, where $T_{Heartbeat}$ is the heartbeat waiting time to check whether the file is updated, T_{read} is the time to read the size of the key file, T_{send} is the heartbeat command sending time, and T_{other} is the heartbeat waiting timeout. In the figure, when the user's heartbeat waiting for the longest time T satisfies $T \geq T_{Heartbeat}$, the system runs normally; otherwise, the system shows that the QKD process is abnormal.

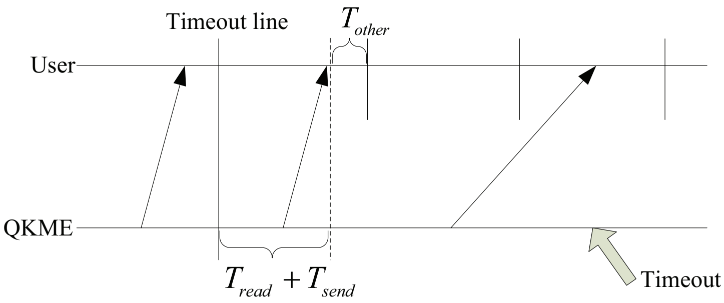

Figure 5. QKD real-time monitoring.

(4) Meeting minutes sharing: After the meeting, user uses the keyword generation algorithm to generate the meeting minutes' keywords. Then, they hash the keyword to obtain the keyword index. The meeting minutes are encrypted with quantum keys to the ciphertext. Together with the quantum key ID used, they are uploaded to the alliance chain as a transaction and broadcast, and other users on the alliance chain verify it. The past video conference minutes' information is stored on the alliance chain for legal users to view. The structure of the alliance chain is shown in Figure 6.

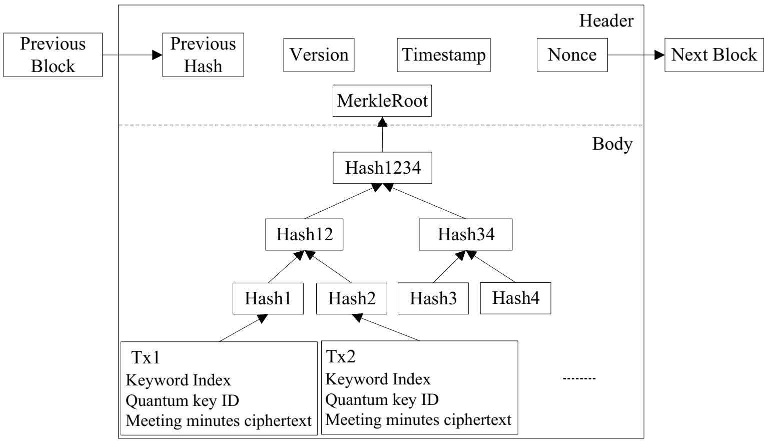

Figure 6. Alliance chain structure.

(5) Meeting minutes query: In order to facilitate users to query meeting minutes multiple times, we propose an efficient query method model based on B+ tree. The model consists of three layers, the data layer, cache layer, and application layer. The data query model is shown in Figure 7.

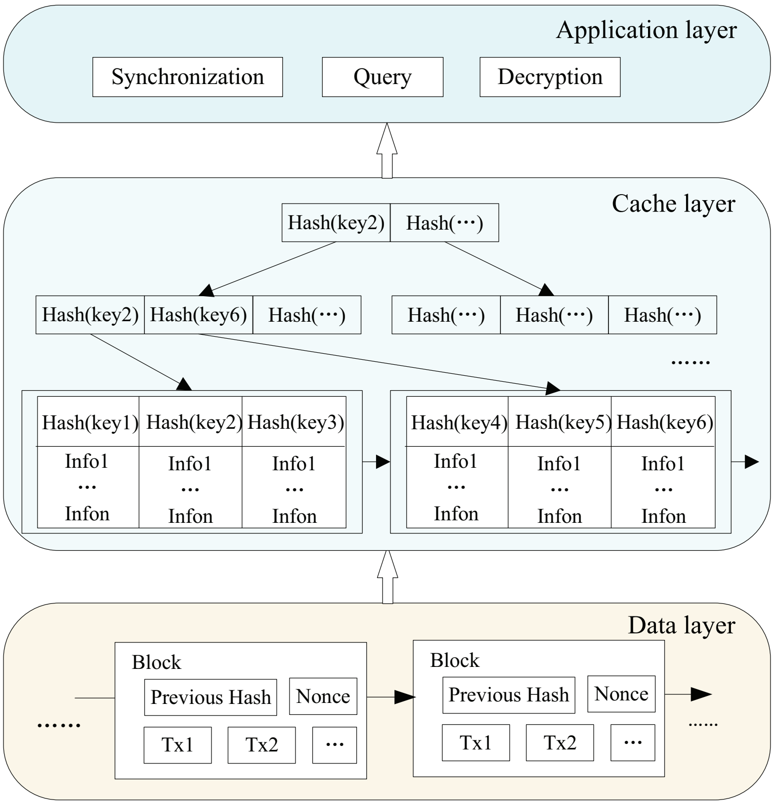

Figure 7. Data query model.

Data layer: The alliance chain platform, the blockchain that stores the ciphertext information of the meeting minutes.

Cache layer: The cache layer reads the ciphertext transaction information of the meeting minutes in each block and caches it in the B+ tree. The transaction information indexed by the same keyword is stored in the same adaptive space.

Application layer: The application layer includes synchronization, query, and decryption operations. Users search for transaction information in the cache layer. If the corresponding data are found, the query result will be returned. Then, through the decryption operation, the plaintext of the meeting minutes is obtained. If no transaction information is found, insert the newly generated transaction information of the blockchain into the B+ tree through synchronous operation and query again.

3.2. Threat Model

There are three main threats in this scheme:

(1) Network monitoring: The attacker knows the user's address information by some means. In the video conference system network, the attacker can intercept the encrypted video information by setting the network interface to the monitoring mode with tools such as Wireshark. The attacker tries to decrypt the video information and obtain the plaintext in order to achieve a certain purpose.

(2) Minutes ciphertext cracking: The attacker knows the hash algorithm and creates a keyword index. Disguised as a legitimate user, the keyword index is sent to the alliance chain platform, and the alliance chain searches the corresponding cryptographic

information according to the index value and returns it to the attacker. The attacker tries to crack the ciphertext information and obtain the plaintext of the minutes.

(3) Unpredictable threats: Quantum states are transmitted through quantum channels, which is easily influenced by the environment, eavesdropping, and other factors, and it is a possible that the key cannot be generated.

3.3. Security Requirements

This scheme mainly has the following three security objectives:

First, during the operation of the video conference system, the network monitor cannot analyze the intercepted video encryption information; second, the attacker cannot decrypt the ciphertext of the meeting minutes returned by the alliance chain; third, the video conference system can run stably under the unpredictable external environment.

In order to achieve the first and second goals, QKD technology is used to replace classical technology, and symmetric quantum keys are generated among users to realize key security sharing and effectively prevent attackers from obtaining keys. The important content and meeting minutes of the video conference are encrypted by the quantum key to prevent attackers from decrypting and obtaining plaintext information.

To ensure the third security goal, this scheme sets up two key encryption schemes of classical and quantum according to the security level of the video conference content. Considering the unpredictability of quantum channel security and the low rate of quantum key generation, quantum encryption includes two schemes of one-time pad + AES to ensure the operation of the system.

4. Our Construction

This section introduces the detailed construction of the model. Based on the three-node star quantum communication network (Bob, Alice1, Alice2), through the quantum key distribution, Bob–Alice1 generates a symmetric quantum key, and Bob–Alice2 generates a symmetric quantum key. The video conference system server is placed at Bob; user1 is in Alice1; user2 is in Alice2. Bob, Alice1, and Alice2 are local area networks, respectively.

4.1. Hybrid Encryption

According to the secret level of the video conference content, the system designs two data encryption schemes: one is the classical key encryption scheme and the other is the quantum key encryption scheme. The two schemes can be switched without affecting the operation of the video services. This configuration is an excellent business redundancy mechanism of quantum secure communication in a video conference system:

(1) The secret level of the video conference content is low, and the business data adopt a classical key encryption scheme.

(2) The secret level of the video conference content is high, and the business data adopt the quantum key encryption scheme.

4.1.1. Encrypt

If user1 (main site) chooses the quantum key encryption scheme, it will be encrypted by the one-time pad encryption algorithm. Once the key amount is insufficient, it will switch to the AES encryption algorithm. If the classical key encryption algorithm is selected, it will be encrypted by AES. User1 transmits the encrypted information to the VCSS; see Algorithm 1 for the specific process.

Algorithm 1: Video encryption algorithm

```
Input: Video plaintext $M$
Output: Video ciphertext $C$
Init $quantum\_AES\_flag = FALSE$;
User1 sets the encryption mode $quantum\_flag$;
Send flag $Server \leftarrow User1(quantum\_flag)$;
if $quantum\_flag == TRUE$ then
    if $quantum\_AES\_flag == FALSE$ then
        for $i = 1$ to $QT_{A1}.len()$ do
            $C_i = En_{xor}(QT_{A1}^{l_i}, M_i)$;
            Send ciphertext $Server \leftarrow User1(C_i)$;
            if Insufficient quantum keys then
                BREAK;
            end
        end
    end
    $quantum\_AES\_flag = TRUE$;
    $ID_{AES} = i$;
    $Server \leftarrow User1(quantum\_AES\_flag, ID_{AES})$;
    $C = En_{AES}(QT_{A1}^{l_i}, M)$;
end
if $quantum\_flag == False$ then
    $C = En_{AES}(ck, M)$;
end
return $C$;
```

4.1.2. Decrypt

The VCSS receives the ciphertext information sent by user1 and decrypts it using the key in table QT_{B1}. Then, use the key in the table QT_{B2} to encrypt and send it to user2. User2 decrypts the ciphertext with the key in the table QT_{A2} and obtains the video plaintext data, and the decryption process of user2 is the same as the decryption process of the server, which is not repeated here. See Algorithm 2 for the server decryption process.

4.2. QKD Real-Time Monitoring

The QKME sets T_{start} as the start time of heartbeat monitoring, $T_{current}$ as the current system time, T_{cread} as the current system time of reading the file size, and T_{csend} as the current system time of sending the heartbeat command. At regular intervals, the QKME sends $T_{Heartbeat}$ to the user, and the user sets the longest waiting time of heartbeat T. If $T \geq T_{Heartbeat}$, the QKD network runs normally; otherwise, the video conference system displays the QKD network abnormality in the main interface. See Algorithm 3 for the specific flow of the real-time monitoring algorithm.

Algorithm 2: Video decryption algorithm

```
Input: Video ciphertext $C$
Output: Video plaintext $M$
Server:;
$User2 \leftarrow Server(quantum\_flag, quantum\_AES\_flag)$;
if $quantum\_flag == TRUE$ then
    if $Quantum\_AES\_flag == FALSE$ then
        for $i = 1$ to $QT_{B1}.len()$ do
            $M_i = De_{xor}(QT_{B1}^{l_i}, C_i)$;
            $C_i = En_{xor}(QT_{B2}^{l_i}, M_i)$;
            Send ciphertext $User2 \leftarrow Server(C_i)$;
        end
    end
    if $Quantum\_AES\_flag == TRUE$ then
        $i = ID_{AES}$;
        $User2 \leftarrow Server(quantum\_AES\_flag, ID_{AES})$;
        $M = De_{AES}(QT_{B1}^{l_i}, C)$;
        $C = En_{AES}(QT_{B2}^{l_i}, M)$;
        Send ciphertext $User2 \leftarrow Server(C_i)$;
    end
end
if $Quantum\_flag == FALSE$ then
    $User2 \leftarrow Server(C)$;
end
User2:;
if $quantum\_flag == TRUE$ then
    if $Quantum\_AES\_flag == FALSE$ then
        for $i = 1$ to $QT_{A2}.len()$ do
            $M_i = De_{xor}(QT_{A2}^{l_i}, C_i)$;
        end
    end
    if $Quantum\_AES\_flag == TRUE$ then
        $i = ID_{AES}$;
        $M = De_{AES}(QT_{A2}^{l_i}, C)$;
    end
end
if $Quantum\_flag == FALSE$ then
    $M = De_{AES}(ck, C)$;
end
return $M$;
```

Algorithm 3: Real-time monitoring algorithm

```
Input: NULL
Output: Heartbeat monitoring results
QKME:;
while $TRUE$ do
    $T_{start} = T_{current}$;
    $T_{read} = T_{cread} - T_{start}$;
    $T_{send} = T_{csend} - T_{cread}$;
    if Increased key file capacity then
        $T_{other} = 3s$;
    else
        $T_{other} = 100s$;
    end
    $T_{Heartbeat} = T_{start} + T_{read} + T_{send}$;
    $User \leftarrow QKME(T_{Heartbeat})$;
    $Sleep(10)$;
end
User:;
while $TRUE$ do
    if $T \geq T_{Heartbeat}$ then
        QKD network normally;
    else
        System displays the QKD network abnormality;
    end
    $Sleep(10)$;
end
return $T_{Heartbeat}$;
```

5. Security Analysis

(1) Data security:

Key security ensures the security of communication data. Existing key distribution schemes generally use assumed secure channels to transmit key system parameters or the two communicating parties can only meet privately to exchange keys. In practical applications, due to the complexity of the network, there are security problems such as eavesdropping or cracking, so the quantum key distribution mechanism is introduced. The system makes full use of the security characteristics of the quantum key distribution and generates a theoretically absolutely secure symmetric key through the quantum channel. Encrypting data through "one-time pad" can effectively prevent monitoring and theft and strengthen the data security of the existing scheme.

Assuming that the attacker knows the user's communication address information, during the operation of the video conference system, the Wireshark tool is used to successfully intercept the network data packet $\{C_0, C_1, C_2 \cdots C_n\}$ of important data, but the encrypted quantum key is private and updated in real-time, so $\{C_0, C_1, C_2 \cdots C_n\}$ is indistinguishable in the form of ciphertext. For example, the attacker constructs a pair of information (m_0, m_1), $m_0, m_1 \in M$, and (m_0, m_1), which is encrypted by quantum key encryption method $En_{opt}(qk, M)$, respectively, where qk is the real-time update by the QKD network. By repeated encryption, the ciphertext (c_i, c_j) obtained by each plaintext (m_0, m_1) is different, that is the attacker does not know which plaintext (m_0, m_1) corresponds to (c_i, c_j). Therefore, it can effectively resist eavesdropping attacks.

Assume that the attacker can access the alliance chain and query the meeting minutes' information through the keyword index. The attacker constructs the keyword kw' through the collision attack of the hash function, calculates $hkw = Hash(kw')$, and uses the query function of the alliance chain to obtain the ciphertext C of the meeting minutes associated with hkw'. Similarly, because the ciphertext C stored in the blockchain is indistinguishable, the attacker cannot decrypt C and obtain the plaintext M of the meeting minutes. At the same time, due to the one-way characteristic of the hash function, the original data cannot be deduced by collision attack, so the video conference communication data and meeting minutes' data are safe.

(2) System stability:

Because of the influence of the channel, environment, eavesdropping, and other factors, quantum keys cannot be generated. In order to ensure the stable operation of the system, two encryption schemes are provided, namely the "one-time pad + AES" quantum key encryption scheme and the "AES" classical key encryption scheme. The two schemes can be switched at any time when the system is running. When the security level of the conference content is high, the user switches the quantum key encryption scheme. When the quantum key is sufficient, the system uses the "one-time pad" encryption scheme; when the quantum key capacity is insufficient, the system will automatically switch to the "AES" encryption scheme, and there will not be a crash. When the security level of the conference content is low, the user can switch to the classical key encryption scheme without consuming quantum keys. At the same time, in order to monitor whether the QKD network is running normally, a heartbeat monitoring mechanism is set to determine $T \geq T_{Heartbeat}$, so that the user can grasp the real-time dynamics of the QKD network and ensure the stable operation of the video conference system.

6. Experimental Evaluation

6.1. System Network Topology

The system combines the time phase quantum key distribution equipment with the classical network, and the designed video conference system network topology based on the quantum key is shown in Figure 8, wherein:

(1) The dotted line indicates the quantum channel for quantum key distribution; the solid line indicates the classic channel for the encryption communication line of the video conference system. The line attenuation value of the main site and the sub-site is less than or equal to 18 dB. All lines are bare fibers, and devices such as optical amplifiers cannot be included in the line.

(2) The video conference system comprises a main site and a sub-site. The main site includes a single-receiving time phase QKD device; the sub-site includes a single-emitting time phase QKD device. The key is generated through the QKD device quantum channel of each site and saved to the quantum key management equipment of each site. Based on the decoy BB84 protocol, the QKD device integrates the quantum signal- emitting or -receiving module with a working frequency up to GHz. The signal light has a wavelength of 1550 nm and a pulse width of 200.

(3) Through wavelength division multiplexing, the multi-channel quantum wavelength division multiplexing terminal aggregates multiple QKD devices into the same fiber for transmission, which achieves multiple expansion of the quantum key distribution rate.

(4) The video conference system consists of one server and two users, namely user1 and user2. The server is a ThinkServer TS540 with a CPU model of Intel Xeon E3-1225, a clock speed of 3.2 GHz, and a memory size of 4 GB. The user uses an 8G, i5-4200U industrial computer. The virtual machine uses the Centos 6.6 64 bit operating system.

(5) Each terminal node can directly access office applications such as voice calls and faxes or access office data subnets. When the internal IP service data performs secure communication with the outside, it needs to be encrypted by a quantum key.

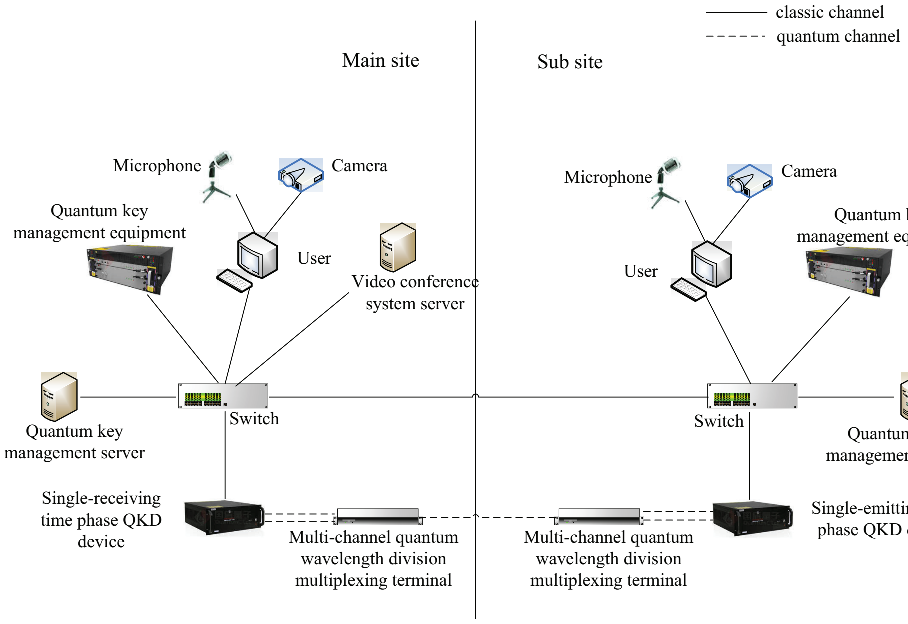
classic channel
quantum channel
Main site
Sub site
Microphone
Camera
Quantum key management equipment
User
Video conference system server
Quantum key management server
Switch
Single-receiving time phase QKD device
Multi-channel quantum wavelength division multiplexing terminal
Microphone
Camera
User
Quantum key management equipment
Switch
Quantum key management server
Single-emitting time phase QKD device
Multi-channel quantum wavelength division multiplexing terminal

Figure 8. The video conference system network topology based on the quantum key.

6.2. Analysis of Quantum Key Generation

The distance between the main site and the branch site is about 33.6 km, and the outdoor temperature is 5 °C. When the QKD network is started, initialization work such as basis comparison is needed. In this process, Alice and Bob choose to keep the part of the data with the same basis and discard the data with different bases. After the basis comparison, Alice and Bob count the number of signal states, decoy states, and vacuum states and then obtain the corresponding detection rate. Let the optical error rate caused by the imperfect optical calibration of the system be e_{opt} and the detection rates of the signal state, decoy state, and vacuum state be S, D, and V, respectively, then the error rates of signal state and decoy state are $e_s = e_{opt} + 0.5 \times {}^{V}/_{S}$, $e_D = e_{opt} + 0.5 \times {}^{V}/_{D}$. The calculation results are shown in Figure 9. QKD runs stably for 50 min; the signal state error rate is about 0.92%, and the decoy state error rate is about 2.07%.

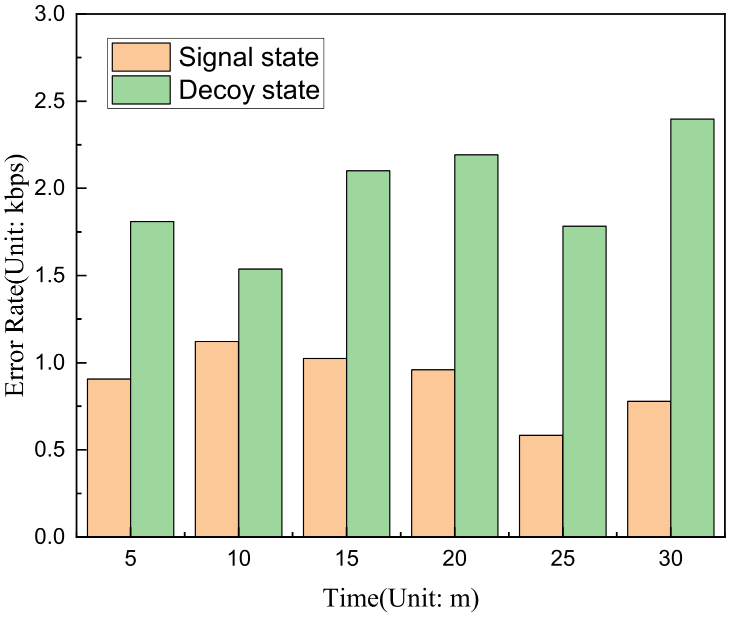

Figure 9. The quantum key error rate.

In the QKD process, Alice and Bob generate different sifted keys due to noise such as channel interference, optical adjustment problems, and dark counting of detectors. As shown in Figure 10, the key rate of sifted keys is about 24.57 kbps. In order to ensure the correct encryption and decryption of data, the sifted keys are corrected. After processing, Alice and Bob generate the same final key. The final key is the symmetric key for information encryption and decryption, and the quantum key rate is about 3.64 kbps. The key rate after error correction is lower than that of the sifted key, and the quantum key is stored in the QKME.

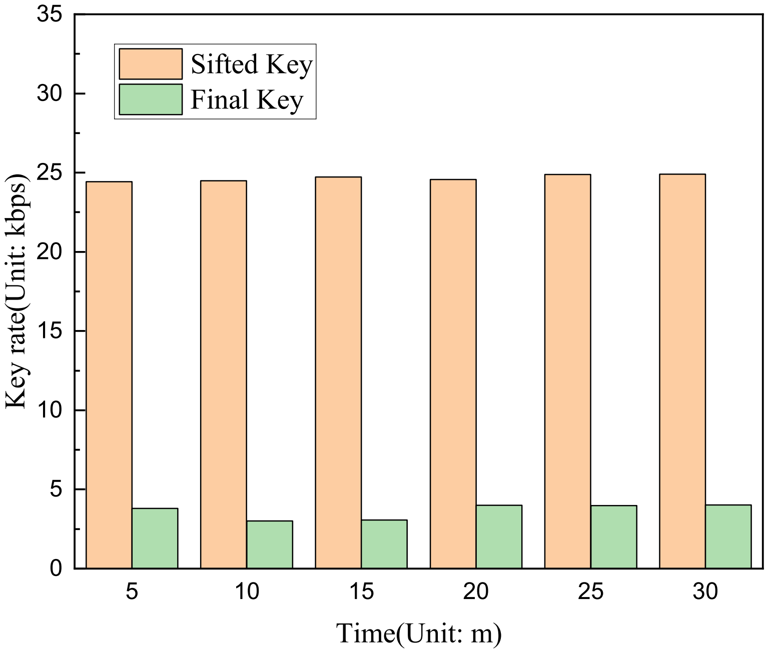

Figure 10. The quantum key rate.

In order to observe the relationship between the quantum key rate and the storage space occupied by the image, this paper lists the memory space occupied by single-frame images with 160 × 120, 320 × 240, 640 × 480, 800 × 600, 1024 × 768, when the color depth is 3, 8, and 16 bits. As shown in Table 2, the image with a resolution of 160 × 120 and a color depth of 3 bits occupies the smallest space. The higher the resolution and the greater the color depth, the larger the space the image takes up. In order to ensure the stable and clear operation of the system, the resolution of this system is 320 × 240. In the quantum key generation, the quantum key rate is about 3.64 kbps, which is far lower than the storage space occupied by the image. Therefore, the QKD network needs to run for a long time and store enough quantum keys.

Table 2. Image size in storage space (unit: KB).

Color Depth	160 × 120	320 × 240	640 × 480	800 × 600	1024 × 768
3	7.03	28.13	112.5	175.78	288
8	18.75	75	300	468.75	768
16	37.5	150	600	937.5	1536

6.3. System Operation Data Analysis

The video conference system consists of a login interface and a main interface. The main interface includes functions such as site layout, classical encrypted communication, and quantum encrypted communication. QKD equipment generates quantum keys in real-time, and the quantum key rate is about 3.64 kbps. According to the key reading speed of 3 kb/s, the two sites are stored in their respective quantum key management servers. At the same time, the main site writes the key to the video conference system server. After the video conferencing system is started, the classic encryption is used by default, and the user of the main conference site switches to use quantum encryption. After running for 300 s, the total key consumption is shown in Figure 11 and Table 3. The key adopts the classical key + quantum key, so the key consumption increases with time. In 300 s of operation, user1 and user2 use about 62.74 KB keys per second, while the reading speed of quantum keys is 3 kb/s, so the algorithm of "one at a time + AES" can ensure the normal operation of the system.

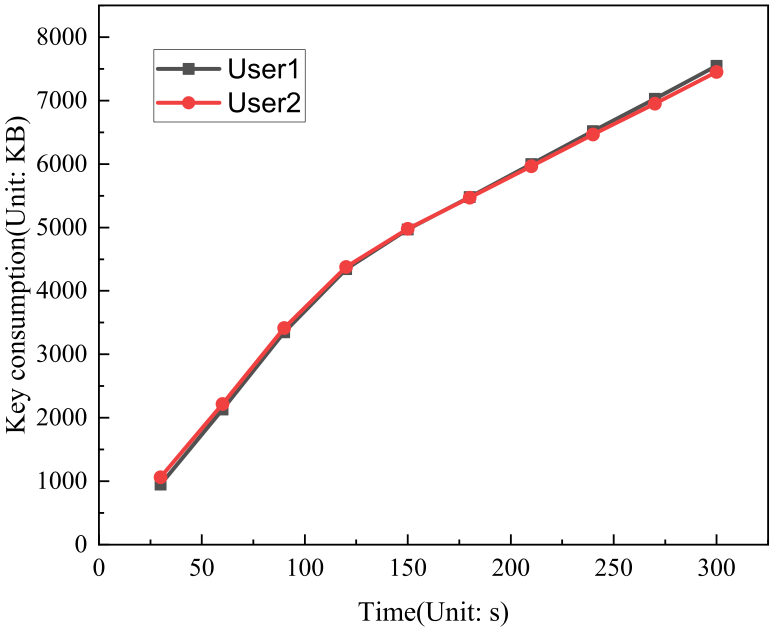

Figure 11. The quantum key consumption of the user.

Table 3. The quantum key consumption in different time periods.

	30 s	60 s	90 s	120 s	150 s	180 s	210 s	240 s	270 s	300 s
User1	948.17	2129.58	3345.83	4345.2	4969.9	5479.06	5998.53	6517.91	7026.44	7548.63
User2	1064	2219.31	3414.83	4379.33	4980.24	5468.71	5966.79	6464.92	6952.69	7453.51

6.4. Scheme Comparison

This scheme is a quantum key distribution network based on actual commercial optical fiber, which realizes the unconditional security scheme of image encryption transmission. The comparison between the scheme and other schemes based on quantum keys for encrypting images is shown in Table 4. References [21,22] used theoretically absolutely secure quantum keys, but the encryption method is only an XOR operation, which does not take into account the case that the key consumption is greater than the key generation for multiple consecutive images. The quantum keys used in References [25,27,28] have pseudo-randomness and are not secure quantum keys generated by a quantum key distribution. At the same time, the scheme in this paper is run in actual commercial fiber, while the image encryption in other references is run in experimental environments.

Table 4. Scheme comparison.

References	Encryption Method	Key Randomness	Key Generation Mode	Scheme Type
[21]	One-time pad	Random	Six-state scheme	Experiment
[22]	One-time pad	Random	BB84 scheme	Experiment
[25]	Arnold scrambling + block encryption + bidirectional diffusion	Pseudo-random	Two-state Pauli quantum walks	Experiment
[27]	Obfuscation method based on back propagation neural network	Pseudo-random	Quantum random walk	Experiment
[28]	AES	Pseudo-random	Quantum random walk	Experiment
Ours	One-time pad + AES	Random	QKD	Actual

In this paper, in order to verify the image encryption and decryption speed, the Lena images were encrypted and decrypted by using an industrial computer with 8G memory and an i5-4200U CPU, a Windows 10 operating system, and MatlabR2016a environment. The experimental results are shown in Table 5. The average encryption and decryption time of Reference [27] is 0.0815 s, and that of Reference [25] is 0.335 s. We adopted a multi-modal hybrid encryption architecture of "one-time encryption" +AES. When the quantum key was sufficient, the one-time encryption algorithm was adopted, and the average encryption and decryption time was 0.0509 s, which is shorter than that in References [25,27]. However, when the quantum key was insufficient, the AES algorithm was adopted, and the average encryption and decryption times were more than 1.5 s. Therefore, in order to run the system quickly, it is necessary to ensure that the quantum key capacity is large enough.

Table 5. Calculation speed test.

	Reference [27]	Reference [25]	Ours
Encryption time (s)	0.075–0.080	0.331–0.339	0.0501–0.0505 or greater than 1.5
decryption time (s)	0.080–0.085	0.331–0.339	0.0510–0.0515 or greater than 1.5

6.5. Execution Cost

In order to test the time cost, the hardware simulation environment for the blockchain application experiment was an Intel Core i5-7200 2.50 GHz dual-core processor with 12G of running memory and a 64 bit Windows 10 computer operating system. The ubuntu20.04 system with 4 GB of memory and 50 GB of hard disk was established by using VMware Workstation Pro 16.2.3. With the open-source Hyperledger fabric v2.2, we built a blockchain network with one Org organization, two peer nodes, and one orderer node and set the block out time to 2 s. The application program was built based on the fabric go sdk.

Considering the actual situation of the video conference system, miners generate blocks after uploading the ciphertext information of the meeting minutes to the blockchain. That is, each block only saves the ciphertext information of one meeting minutes. We tested the time cost of uploading the meetings' minutes to the blockchain with different lengths, as shown in Figure 12. According to the length of the meeting minutes in real life, the user's application will use 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000 characters as the ciphertext information of the meeting minutes and upload them to the blockchain, and the average block generation time is 2232 milliseconds. We found that the block generation time of 300 characters was the longest, while that of 1000 characters was the shortest, so the block generation time had little relationship with the character length, but it was related to the network delay between the application and the blockchain. The reason for the network delay in our experiment was probably the processing power of the computer at that time. According to this experiment, based on the processing time of 1000 characters, the network delay times of other characters were 83, 24, 191, 0, 98, 3, 0, 169, and 34, respectively.

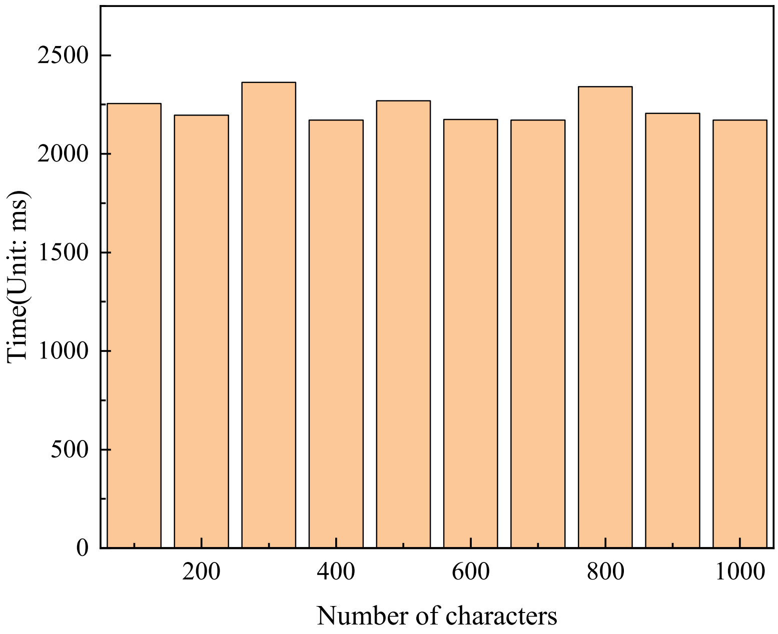

Figure 12. Generate block time with different character lengths.

We created 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000 blockchain blocks, respectively, for the simulated query experiments and compared the B+ tree query of the cache layer with the original query efficiency of the blockchain, as shown in Figure 13 and Table 6. The query time of B+ tree is the return time of the user's cache layer according to the keyword index. The original query time of the blockchain is the time when the user sends the keyword index to the blockchain and the blockchain returns to the user after querying. In order to ensure the correctness, we calculated the time consumption of different query algorithms, respectively, and then, took the average of 10 times. Through calculation, the original query times of the blockchain with different lengths were 18.9, 19.2, 20.4, 19, 19.6, 21.5, 21.9, 20.3, 17.4, and 20.3, respectively; the query time of the cache B+ tree was 5.7, 8.1, 4.8, 4.6, 8.6, 5.9, 5.6, 6.3, 5.1, and 7.3, respectively. From the experimental results, it can be seen that the query efficiency of using the cache layer B+ tree was 3.15-times higher than the original query efficiency of the blockchain, and the query efficiency has obvious advantages.

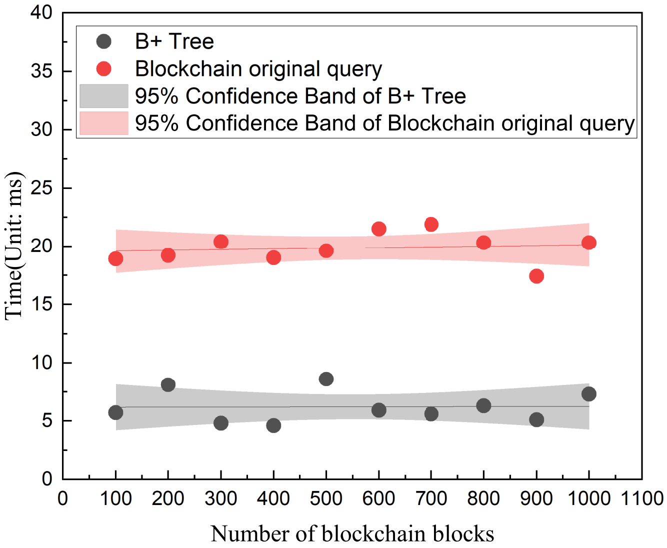

Figure 13. Query time comparison.

Table 6. Query time of different numbers of blockchain blocks.

	100	200	300	400	500	600	700	800	900	1000
B+ Tree	5.7	8.1	4.8	4.6	8.6	5.9	5.6	6.3	5.1	7.3
Blockchain original query	18.9	19.2	20.4	19	19.6	21.5	21.9	20.3	17.4	20.3

7. Conclusions

The security of video conference systems is very important, and the content of the conference often involves military intelligence, commercial secrets, state secrets, and personal privacy. To solve the problem of communication data security, this paper designed a hybrid encryption scheme for quantum secure video conferencing combined with blockchain, making full use of the characteristics and advantages of the quantum key, integrating the quantum key distribution network with the classical network and using classical and quantum double-encryption schemes. At the same time, two quantum encryption schemes of "one-time pad" and AES were adopted to solve the problem of the low key rate of quantum keys. The system implements the QKD network operation monitoring scheme, enabling users to better grasp the QKD network operation. With the advantages of blockchain technology, such as tamper-proofness, decentralization, and traceability, the ciphertext of the meeting minutes is stored in the blockchain, so that the brief content of the meeting can be reviewed. Through the actual operation of the video conference system, the results show that the system has high stability and practicability. A cache-efficient query method based on B+ tree was designed, which was 3.15-times more efficient than the original query of the blockchain. In the future work, we plan to design a high-efficiency quantum key expansion algorithm. Since the video conference uses about 62.74 KB keys per second, the quantum key is expanded to 63 times, which can satisfy the requirements of encrypting and decrypting video conference data.

Author Contributions: Conceptualization, J.Z. and L.S.; methodology, J.Z. and D.Z.; software, J.W. and H.Z.; validation, J.Z. and L.S.; writing—original draft preparation, D.Z.; writing—review and editing, J.Z. and D.Z.; supervision, N.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by Jilin Development and Reform Commission Construction Fund (2020C020-2), the Education Department of Jilin Province with Grants No. JJKH20200577KJ, the Project Plan of Science and Technology Development Center of the Ministry of Education (No. 2020ITA07031) and the Jilin Science and Technology Development Plan Project (Project No. 20210201083GX).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest

Appendix A

Appendix A.1. Quantum Key Storage

In this scheme, Bob stores the quantum key files F_{B1} and F_{B2} corresponding to Alice1 and Alice2; Alice1 stores the symmetric key file F_{A1} with Bob; Alice2 stores the symmetric key file F_{A2} with Bob, wherein $F_{B1} = F_{A1}$, $F_{B2} = F_{A2}$; the keys of Alice1 and Alice2 are different. The VCSS creates the key table QT_{B1} and QT_{B2}; user1 creates the key table QT_{A1}, and user2 creates the key table QT_{A2}. $QT = (l_1, l_2, l_3, \cdots)$; the length of l_i is 1024 Byte. After the quantum key distribution network is normally started, the quantum key is generated in real-time. The keys qk_{BA1} and qk_{BA2} are stored in Bob's F_{B1} and F_{B2} files, respectively. F_{A1} of Alice1 stores qk_{BA1}, and F_{A2} of Alice2 stores qk_{BA2}. The function of

Algorithm A1 is to read the key in file F and write it into the database table of the VCSS, user1, and user2.

Algorithm A1: Quantum key is stored to the database table

```
Input: Quantum key binary string
Output: NULL
$k = 1$;
while $TRUE$ do
    for $i = 1$ to $3$ do
        for $j = 1$ to $1024$ do
            $QT^{I_k}_{B1} = F_{B1}[j]$;
            $QT^{I_k}_{B2} = F_{B2}[j]$;
            $QT^{I_k}_{A1} = F_{A1}[j]$;
            $QT^{I_k}_{A2} = F_{A2}[j]$;
        end
        $k = k + 1$;
    end
    $Sleep(1)$;
end
$return$;
```

Appendix A.2. Meeting Minutes' Sharing

In the meeting minutes' data query model, in order to improve the efficiency of multiple queries by users, the blockchain transaction information is cached in the B+ tree. If the ciphertext information of the meeting minutes is indexed by the same keyword, increase the memory space of the leaf node according to the ciphertext length and save it to the same leaf node. The blockchain data are updated after each meeting, so the user needs to perform a synchronization operation to insert the newly generated transaction information of the blockchain into the B+ tree. The user searches the B+ tree of the buffer area according to the keyword index. According to the search principle of B+ tree, the leaf node associated with the keyword index is retrieved, and the ciphertext information of the meeting minutes is taken out. The information includes the quantum key ID and the ciphertext of the meeting minutes. Using the quantum key ID as the query condition in the database, then the decryption operation is performed with the meeting ciphertext to obtain the plaintext of the meeting, which realizes the sharing of meeting minutes. See Algorithm A2 for the specific process.

Algorithm A2: Meeting minutes' sharing algorithm

```
Input: Meeting minutes' plaintext $M$
Output: Shared meeting minutes' plaintext $M$
User1:;
$kw = GetKewWord(M)$ # Create keyword functions;
$hkw = Hash(kw)$ #Keyword Index;
$C = En_{AES}(QT_{A1}^{l_{id1}}, M)$;
$QT_{A1}^{l_{id1}}$ save to Minutes key table;
Package $C$, keys $id1$ and $hkw$, and upload the blockchain;
Server:;
$C' = En_{xor}(QT_{A2}^{l_{id2}}, QT_{A1}^{l_{id1}})$;
$User2 \leftarrow Server(C')$;
User2:;
$QT_{A1}^{l_{id1}} = De_{xor}(QT_{A2}^{l_{id2}}, C')$;
$QT_{A1}^{l_{id1}}$ save to Minutes key table;
User2 query $hkw2$;
B+ tree is generated by blockchain records;
if $hkw == hkw2$ then
    $GetC, id1$;
    $M = De_{AES}(QT_{A2}^{l_{id1}}, C)$;
end
$return\ M$;
```

References

1. Wen, T.; Zhang, Q.B.; An, W.T. Cross-region technology video conference network construction based on VPN technology. *J. Liaoning Univ. Technol. Sci. Ed.* **2019**, *39*, 164–168.
2. Cheng, H.X.; Zhang, L.; Zou, D. Research on dynamic resource reservation algorithm for enterprise video conferencing based on variable bandwidth. *J. Commun.* **2018**, *39*, 213–219.
3. Liao, Z.W.; Zhang, L.; Deng, J.F. Research on framework for enterprise video conference system based on SDN. *J. Huazhong Univ. Sci. Technol. Sci. Ed.* **2016**, *44*, 204–209.
4. Liao, X.F. Research on fast chaotic encryption algorithm applied to multimedia video files. *Mod. Electron. Tech.* **2019**, *42*, 100–102.
5. Zhu, Y.P. Video encryption algorithm combining wavelet transform with chaos keying. *Comput. Appl. Softw.* **2019**, *36*, 311–316.
6. Di, X.Q.; Wang, Y.Z.; Li, J.Q.; Cong, L.G.; Qi, H. Video encryption method based on hyperchaos of quantum cellular neural networks. *J. Jilin Univ. (Eng. Technol. Ed.)* **2018**, *48*, 919–928.
7. Eid, M.M.; El-kenawy, E.S.M.; Ibrahim, A. A new hybrid video encryption technique based on chaos cryptography. *J. Comput. Sci. Inf. Syst.* **2021**, *2*, 1–8.
8. Kordov, K.; Dimitrov, G. A new symmetric digital video encryption model. *Cybern. Inf. Technol.* **2021**, *21*, 50–61. [CrossRef]
9. Alawi, A.R.; Hassan, N.F. A proposal video encryption using light stream algorithm. *Eng. Technol. J.* **2021**, *39*, 184–196. [CrossRef]
10. Cheng, S.L.; Wang, L.J.; Ao, N.X.; Han, Q. A selective video encryption scheme based on coding characteristics. *Symmetry* **2020**, *12*, 332. [CrossRef]
11. Mohseni, M.; Read, P.; Neven, H.; Boixo, S.; Denchev, V.; Babbush, R.; Fowler, A.; Smelyanskiy, V.; Martinis, J. Commercialize quantum technologies in five years. *Nature* **2017**, *543*, 171–174. [CrossRef] [PubMed]
12. Wang, Y.L.; Xu, Q.L. Principle and research progress of quantum computation and quantum cryptography. *J. Comput. Res. Dev.* **2020**, *57*, 2015–2026.
13. Bennett, C.H.; Brassard, G. Quantum cryptography: Public key distribution and coin tossing. In Proceedings of the International Conference on Computers, Systems & Signal Processing, Bangalore, India, 9–12 December 1984; pp. 175–179.
14. Guo, G.C.; Zhang, H.; Wang, Q. Review on development of quantum information technology. *J. Nanjing Univ. Posts Telecommun. (Natural Sci. Ed.)* **2017**, *37*, 1–14.
15. Tang, Z.F.; Qin, Y.Y.; Jiang, Z.M.; Krawec, W.; Zhang, P. Quantum-secure networked microgrids. In Proceedings of the 2020 IEEE Power & Energy Society General Meeting, Montreal, Canada, 3–6 August 2020; pp. 1–5.
16. Banerjee, S. Indian quantum communication enabled space security. *Indian Sci. Cruiser* **2021**, *35*, 58–62.
17. Yan, L.C.; Chen, Z.Y.; Yu, X.H.; Lyu, Q.; Zhu, J.; Zhao, Z.Y. Security interaction framework for electricity service in new-type town based on quantum key distribution. *Autom. Electr. Power Syst.* **2020**, *44*, 28–35.

18. Chen, Z.; Gao, D.; Wang, D.; Li, G.; Ge, B.; Zhao, Z. Quantum key based optimal data protection model for power business. *Autom. Electr. Power Syst.* **2018**, *42*, 115–121.
19. Chen, Z.Y.; Gao, D.Q.; Wang, D.; Guochun, L.; Xiaojing, W. Performance evaluation of power quantum secure communication system for energy internet. *J. Comput. Res. Dev.* **2017**, *54*, 711–719.
20. Goggin, M.E.; Sundaram, B.; Milonni, P.W. Quantum logistic map. *Phys. Rev. A* **1990**, *41*, 5705–5708. [CrossRef]
21. Wang, P.Z.; Wang, Y.L.; Yin, Z.H.; Cheng, M.T. An image encryption algorithm based on six-state quantum key. *J. Anhui Univ. Technol. Sci.* **2014**, *31*, 303–308.
22. Zhang, K.; Gao, H.X. A new image encryption algorithm based on quantum key and chaotic mapping. *J. Anhui Univ. Technol. Sci.* **2016**, *33*, 167–171.
23. Abd-El-Atty, B.; El-Latif, A.; Ahmed, A.; Venegas-Andraca, S.E. An encryption protocol for neqr images based on one-particle quantum walks on a circle. *Quantum Inf. Process.* **2019**, *18*, 272–297. [CrossRef]
24. Abd-El-Atty, B.; Iliyasu, A.M.; El-Latif, A. A multi-image cryptosystem using quantum walks and chebyshev map. *Complexity* **2021**, *2021*, 9424469. [CrossRef]
25. Yang, X.G.; Wang, X.X. Image encryption algorithm based on the 2-state Pauli quantum walks. *J. Anhui Univ. (Natural Sci. Ed.)* **2021**, *45*, 37–48.
26. Wang, Y.N.; Song, Z.Y.; Ma, Y.L.; Hua, N.; Ma, H.Y. Color image encryption algorithm based on DNA code and alternating quantum ran-dom walk. *Acta Phys. Sin.* **2021**, *70*, 230302. [CrossRef]
27. Ge, B.; Luo, H.B. Image encryption application of chaotic sequences incorporating quantum keys. *Int. J. Autom. Comput.* **2020**, *17*, 123–138. [CrossRef]
28. Liu, G.; Li, W.; Fan, X.; Li, Z.; Wang, Y.; Ma, H. An Image Encryption Algorithm Based on Discrete-Time Alternating Quantum Walk and Advanced Encryption Standard. *Entropy* **2022**, *24*, 608. [CrossRef]
29. Gao, Y.L.; Chen, X.B.; Xu, G.; Yuan, K.G.; Liu, W.; Yang, Y.X. A novel quantum blockchain scheme base on quantum entanglement and DPoS. *Quantum Inf. Process.* **2020**, *19*, 1–15. [CrossRef]
30. Wen, X.J.; Chen, Y.Z.; Fan, X.C.; Yi, Z.Z.; Jiang, Z.L.; Fang, J.B. Quantum blockchain system. *Mod. Phys. Lett. B* **2021**, *35*, 2150343. [CrossRef]
31. Abulkasim, H.; Mashatan, A.; Ghose, S. Quantum-based privacy-preserving sealed-bid auction on the blockchain. *Optik* **2021**, *242*, 167039. [CrossRef]
32. Antal, C.; Cioara, T.; Antal, M.; Anghel, I. Blockchain platform for COVID-19 vaccine supply management. *IEEE Open J. Comput. Soc.* **2021**, *2*, 164–178. [CrossRef]
33. Arndt, T.; Guercio, A. Blockchain-based transcripts for mobile higher-education. *Int. J. Inf. Educ. Technol.* **2020**, *10*, 84–89. [CrossRef]
34. Thamrin, R.M.H.; Harahap, E.P.; Khoirunisa, A.; Faturahman, A.; Zelina, K. Blockchain-based land certificate management in indonesia. *ADI J. Recent Innov.* **2021**, *2*, 232–252. [CrossRef]
35. Xu, W.Y.; Wu, L.; Yan, Y.X. Privacy-preserving scheme of electronic health records based on blockchain and homomorphic encryption. *J. Comput. Res. Dev.* **2018**, *55*, 2233–2243.
36. Yang, X.D.; Xi, W.T.; Wang, J.Q.; Chen, A.; Wan, C. Electronic evidence sharing scheme of Internet of vehicles based on signcryption and blockchain. *J. Commun.* **2021**, *42*, 236–246.
37. Tan, H.B.; Zhou, T.; Zhao, H.; Zhao, Z.; Wang, W.D.; Zhang, Z.X.; Sheng, N.Z.; Li, X.F. Archival Data Protection and Sharing Method Based on Blockchain. *J. Softw.* **2019**, *30*, 2620–2635.