Data Storage Method, Data Query Method and Apparatuses

Cross Reference to Related Application

[1] The present application is based on and claims priority to Chinese Patent Application No. 201710910293.0, filed on September 29, 2017, which is incorporated herein by reference in its entirety.

Background

[2] In the information age, data have values. For example, if a bank could acquire house purchase records of a client from a house sales center, the bank would be able to better assess the client's credit; if the house sales center can acquire clients' credit scores from the bank, then different sales strategies could be employed for different clients. In the example above, there is apparently a demand for sharing each other's data between the bank and the house sales center. In fact, more and more institutions desire to share data as the value of data becomes increasingly significant.

[3] At present, a data sharing platform is typically used as an intermediary. On one hand, the data sharing platform receives and stores data uploaded by various institutions, and on the other hand, the data sharing platform provides data query services to the institutions. Moreover, as not every institution may have a permission to query all data on the data sharing platform (for example, an institution without a finance license cannot legally query data uploaded by finance institutions; for another example, one institution may enter an agreement only with certain other institutions for querying each other's uploaded data), the data sharing platform further needs to set up an access permission for each institution, such that the data query by the each institution on the data sharing platform complies with legal rules or agreements among institutions. Data sharing among institutions is realized in such a manner.

[4] On the basis of the prior art, it is desired to develop safer and more efficient data storage methods and data query methods.

Summary

[5] Embodiments of the present description provide a data storage method, a data query method and apparatuses to solve the problems that existing data access methods cannot ensure data security and have low data query efficiency.

[6] To solve the above technical problems, the embodiments of the present description are achieved as follows:

[7] A data storage method according to some embodiments of the present description comprises: sending, by a blockchain node associated with a blockchain, data to an encryption

device to cause the encryption device to encrypt the data and return the encrypted data to the blockchain node; receiving the encrypted data returned by the encryption device; sending the encrypted data to other blockchain nodes associated with the blockchain to cause the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[8] A data encryption method according to some embodiments of the present description comprises: receiving, by an encryption device, data sent by a blockchain node; encrypting a part of the data; returning the encrypted data to the blockchain node to cause the blockchain node to send the encrypted data to other blockchain nodes for consensus verification and then cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[9] A data query method according to some embodiments of the present description comprises: determining, by a blockchain node associated with a blockchain, data to be queried in encrypted data stored in the blockchain; determining, through a smart contract, whether the blockchain node has a permission to decrypt the data to be queried; if the blockchain node has the permission, sending the data to be queried to an encryption device through the smart contract to cause the encryption device to decrypt the data to be queried and return data obtained from the decryption to the blockchain node; if the blockchain node does not have the permission, refusing to send the data to be queried to the encryption device. A smart contract is a computer protocol intended to digitally facilitate, verify, or enforce the negotiation or performance of a contract. Smart contracts allow the performance of credible transactions without third parties. These transactions may be trackable and irreversible.

[10] A data storage apparatus according to some embodiments of the present description comprises: a first sending module configured to send data to an encryption device to cause the encryption device to encrypt the data and return the encrypted data to the blockchain node; a receiving module configured to receive the encrypted data returned by the encryption device; a second sending module configured to send the encrypted data to other blockchain nodes to cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[11] A data encryption apparatus according to some embodiments of the present description comprises: a receiving module configured to receive data sent by a blockchain node associated with a blockchain; an encrypting module configured to encrypt a part of the data; a returning module configured to return the encrypted data to the blockchain node to cause the

blockchain node to send the encrypted data to other blockchain nodes associated with the blockchain for consensus verification and then cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[12] A data query apparatus implemented by a blockchain node associated with a blockchain according to some embodiments of the present description comprises: a determining module configured to determine data to be queried in encrypted data stored in the blockchain; a determining and processing module configured to determine, through a smart contract, whether the apparatus has a permission to decrypt the data to be queried; if the blockchain node has the permission, send the data to be queried to an encryption device through the smart contract to cause the encryption device to decrypt the data to be queried and return data obtained from the decryption to the apparatus; if the blockchain node does not have the permission, refuse to send the data to be queried to the encryption device.

[13] A data storage device implemented on a blockchain node associated with a blockchain according to some embodiments of the present description comprises one or more processors and memories, wherein the memories store programs, and is configured to execute the following steps by the one or more processors: sending data to an encryption device to cause the encryption device to encrypt the data and return the encrypted data to the blockchain node; receiving the encrypted data returned by the encryption device; sending the encrypted data to other blockchain nodes associated with the blockchain to cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[14] A data encryption device according to some embodiments of the present description comprises one or more processors and memories, wherein the memories store programs, and is configured to execute the following steps by the one or more processors: receiving data sent by a blockchain node associated with a blockchain; encrypting a part of the data; returning the encrypted data to the blockchain node to cause the blockchain node to send the encrypted data to other blockchain nodes associated with the blockchain for consensus verification and then cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[15] According to another aspect, a system implementable as a blockchain node associated with a blockchain comprises a processor and a non-transitory computer-readable storage medium storing instructions that, when executed by the processor, cause the processor to

perform a data storage method, the method comprising: sending data to an encryption device to cause the encryption device to encrypt the data and return the encrypted data to the blockchain node; receiving the encrypted data returned by the encryption device; and sending the encrypted data to other blockchain nodes associated with the blockchain to cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[16] In some embodiments, the method further comprises: determining a first value corresponding to the data through a smart contract, the first value being used to represent a value added into an account of the blockchain node; and sending the blockchain node's own node identifier and the first value to other blockchain nodes to cause the node identifier and the first value to be stored in the blockchain after a consensus verification on the node identifier and the first value by the blockchain nodes passed.

[17] In some embodiments, the encryption device is caused to encrypt a part of the data.

[18] In some embodiments, the method further comprises: determining, by one of the blockchain nodes associated with the blockchain, data to be queried in encrypted data stored in the blockchain; determining, through a smart contract, whether the one blockchain node has a permission to decrypt the data to be queried; if the one blockchain node has the permission, sending the data to be queried to an encryption device through the smart contract to cause the encryption device to decrypt the data to be queried and return data obtained from the decryption to the one blockchain node; and if the one blockchain node does not have the permission, refusing to send the data to be queried to the encryption device.

[19] In some embodiments, each encrypted data stored in the blockchain comprises an unencrypted part; and the determining, by one of the blockchain nodes, data to be queried in encrypted data stored in a blockchain comprises: determining, by the one blockchain node, a data query condition; determining, through the smart contract and according to the data query condition, whether the unencrypted part comprised in each encrypted data stored in the blockchain satisfies the data query condition; if the data query condition is satisfied, using the encrypted data containing the unencrypted part that satisfies the data query condition as data to be queried; and if the data query condition is not satisfied, not using any encrypted data as data to be queried.

[20] In some embodiments, when the one blockchain node has a permission to decrypt the data to be queried, the method further comprises: determining, through the smart contract, a second value corresponding to the data returned by the encryption device, the second value

being used to represent a value deducted from an account of the one blockchain node; and sending, through the smart contract, the one blockchain node's node identifier and the second value to other blockchain nodes associated with the blockchain to cause each of the other blockchain nodes to store the node identifier and the second value in the blockchain after performing consensus verification on the node identifier and the second value with success.

[21] In some embodiments, when the one blockchain node has the permission to decrypt the data to be queried, the method further comprises: determining, through the smart contract, a third value corresponding to the data and node identifiers of other blockchain nodes sharing the data, the third value being used to represent a value added into accounts of the other blockchain nodes sharing the data; and sending, through the smart contract, the determined node identifiers and the third value to the other blockchain nodes to cause each of the other blockchain nodes to store the determined node identifiers and the third value in the blockchain after performing consensus verification on the determined node identifiers and the third value with success.

[22] In some embodiments, the method further comprises: sending, by one of the blockchain nodes associated with a blockchain, a refill request to a refill device to cause the refill device to determine a refill certificate according to the refill request and return the refill certificate to the one blockchain node; the refill certificate comprising a fourth value and the fourth value being used to represent a value added into an account of the one blockchain node; and sending the one blockchain node's node identifier and the refill certificate to other blockchain nodes associated with the blockchain to cause each of the other blockchain nodes to store the node identifier and the fourth value in the blockchain after performing consensus verification on the node identifier and the refill certificate with success.

[23] According to another aspect, a non-transitory computer-readable storage medium storing instructions that, when executed by a processor of a blockchain node, cause the processor to perform a data storage method, the method comprising: sending data to an encryption device to cause the encryption device to encrypt the data and return the encrypted data to the blockchain node; receiving the encrypted data returned by the encryption device; and sending the encrypted data to other blockchain nodes to cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[24] A data query device according to some embodiments of the present description comprises one or more processors and memories. The memories store programs, which, when

executed by the one or more processors, cause the data query device to perform the following steps: determining data to be queried in encrypted data stored in a blockchain; determining, through a smart contract, whether the blockchain node has a permission to decrypt the data to be queried; if the blockchain node has the permission, sending the data to be queried to an encryption device through the smart contract to cause the encryption device to decrypt the data to be queried and return data obtained from the decryption to the blockchain node; if the blockchain node does not have the permission, refusing to send the data to be queried to the encryption device.

[25] From the above technical solutions according to the embodiments of the present description, it can be seen that a blockchain network is formed with each institution acting as a blockchain node in some embodiments of the present description. On one hand, any blockchain node can send its own data to an encryption device for encryption, receive encrypted data returned by the encryption device, and then broadcast the encrypted data to other blockchain nodes to cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success; on the other hand, with a smart contract preset in the blockchain, encrypted data within the query permission scope of the blockchain node can be sent to the encryption device for decryption, namely the blockchain node can query data within its permission. According to some embodiments of the present description, first, data shared by blockchain nodes are stored in a distributed manner in a blockchain maintained by the blockchain nodes, and consequently it is difficult to tamper with the data stored in the blockchain, leading to high security; second, the data are encrypted before being stored in the blockchain, and if a blockchain node wants to acquire data, it needs to have its permission verified via a smart contract, and encrypted data that the blockchain node has a permission to acquire will be sent to an encryption device for decryption and ultimately returned to the blockchain node; third, each blockchain node can rapidly acquire data from the blockchain maintained by itself, which improves the efficiency of data acquisition.

Brief Description of the Drawings

[26] To more clearly describe technical solutions of the present description or the prior art, the accompanying drawings to be used in the description of embodiments or the prior art will be described briefly below. Apparently, the accompanying drawings described below are merely some embodiments of the present description. A person skilled in the art can further obtain other drawings according to these drawings without inventive effort.

[27] FIG. 1 is a flow chart of a data storage method according to some embodiments of the present description;

[28] FIG. 2 is a flow chart of a data encryption method according to some embodiments of the present description;

[29] FIG. 3 is a flow chart of a data query method according to some embodiments of the present description;

[30] FIG. 4 is a schematic diagram of an architecture of a data storage and query system according to some embodiments of the present description;

[31] FIG. 5 is a schematic diagram of a data storage apparatus according to some embodiments of the present description;

[32] FIG. 6 is a schematic diagram of a data encryption apparatus according to some embodiments of the present description;

[33] FIG. 7 is a schematic diagram of a data query apparatus according to some embodiments of the present description;

[34] FIG. 8 is a schematic diagram of a data storage device according to some embodiments of the present description;

[35] FIG. 9 is a schematic diagram of an encryption device according to some embodiments of the present description;

[36] FIG. 10 is a schematic diagram of a data query device according to some embodiments of the present description.

Detailed Description

[37] As described in the Background, an existing data sharing platform can, on one hand, receive and store data uploaded by various institutions, and on the other hand, provide data query services to the institutions. However, such a manner has two problems: one, since the data shared by the institutions are stored by a centralized data sharing platform, data queried by the institutions from the platform would become untrustworthy if someone maliciously tampers with the data stored on the platform; two, the data sharing platform often simultaneously receives data query requests sent from multiple institutions, while the data sharing platform has a limited capacity to process data query requests (it is necessary to perform permission verification on each institution requesting data query). As such, some institutions need to wait in a queue, the data query takes a relatively long time, and the efficiency of data query is low.

[38] In some embodiments of the present description, first, a blockchain network is formed

with each institution acting as a blockchain node, and data shared by the blockchain nodes can be stored, after consensus verification by the blockchain nodes is passed, in the blockchain respectively maintained by the blockchain nodes. In such a manner, it would be difficult to tamper with or lose data stored in the blockchain. Second, if a blockchain node wants to query data shared by other blockchain nodes, the blockchain node can simply query the data stored in the blockchain maintained by itself (the data stored in the blockchain maintained by the blockchain nodes are synchronized), and the efficiency of data query can be improved. Third, to manage data query permission of the blockchain nodes, data are encrypted by an encryption device before being stored in the blockchain. Any blockchain node can acquire the encrypted data directly from the blockchain. After the blockchain node passes a permission check on itself via a smart contract, the encrypted data will be sent to the encryption device for decryption and the blockchain node will ultimately receive the plaintext data returned by the encryption device.

[39] To enable a person skilled in the art to better understand the technical solutions of the present description, the technical solutions in the embodiments of the present description will be clearly and completely described below with reference to the accompanying drawings in one or more embodiments of the present description. Apparently, the embodiments described are merely some, rather than all, of embodiments of the present description. All other embodiments obtainable by a person skilled in the art without inventive effort on the basis of the embodiments in the present description shall fall within the scope of the present description.

[40] The technical solutions in the embodiments of the present description will be described in detail below with reference to the accompanying drawings.

[41] FIG. 1 is a flow chart of a data storage method according to some embodiments of the present description, comprising the following steps:

[42] S100: sending, by a blockchain node associated with a blockchain, data to an encryption device.

[43] The flow shown in FIG. 1 is a flow in which a blockchain node stores its own data into a blockchain (i.e., a flow of sharing data).

[44] In some embodiments of the present description, the blockchain node can be a server or a terminal, and in one example, it can be a server of an institution that shares its own data (in the description below, the blockchain node can also be a server of an institution that requests to query data). For example, if a housing rental company wants to share a blacklist of clients

in arrears with rent with other institutions, a server of the housing rental company can access a blockchain as a blockchain node and send the blacklist to an encryption device.

[45] In some embodiments of the present description, the encryption device is not a blockchain node, but a device for performing encryption and decryption on data and providing public services to all blockchain nodes. In some embodiments, the encryption device can use a symmetric encryption algorithm to encrypt received data and to decrypt received encrypted data.

[46] It should be noted that, as a third-party device, the encryption device has the function to use a particular encryption algorithm to encrypt received data and to use a decryption algorithm symmetric to the encryption algorithm to decrypt received encrypted data, while no blockchain node is able to learn the encryption algorithm used by the encryption device for encryption and decryption. In other words, no blockchain node can perform decryption without permission on encrypted data stored in the blockchain and maintained by itself.

[47] In summary, if a blockchain node wants to share its own data in some embodiments of the present description, the blockchain node can send the data to an encrypted device to cause the encryption device to encrypt the received data and to return the encrypted data to the blockchain node.

[48] S102: receiving the encrypted data returned by the encryption device.

[49] S104: sending the encrypted data to other blockchain nodes associated with the blockchain.

[50] After a blockchain node sends the encrypted data to other blockchain nodes, the blockchain nodes can perform consensus verification on the encrypted data. If the consensus verification is passed, the blockchain nodes save the encrypted data in the blockchain.

[51] In some embodiments, the consensus verification performed on the encrypted data is actually to perform consensus verification, according to a signature of a blockchain node, on whether an identity and encrypted data of the blockchain node has been tampered with. In one example, the blockchain node generates a summary of encrypted data according to the encrypted data, generates a signature according to the summary by using a signature algorithm (a private key of the blockchain node), and then sends the encrypted data and the signature to other blockchain nodes. Each of the other blockchain nodes that receive the encrypted data and the signature will, on one hand, use a public key of the blockchain node to decrypt the signature (if the decryption is successful, it indicates that the signature is truly generated by the blockchain node using its private key) to obtain the summary; on the other

hand, generate a summary according to the received encrypted data, and compare the summary obtained from the decryption and the generated summary. If the comparison result is consistent, it indicates that the encrypted data has not been tampered with during the transmission process. The consensus verification method above is a common legitimacy verification in the field of blockchain technologies and will not be further described.

[52] The reason why it is necessary to send encrypted data (rather than original data) to other blockchain nodes is because a blockchain node may not necessarily have a permission to query all data shared by other blockchain nodes. To control data query permissions of the blockchain nodes, data shared by a blockchain node has to be encrypted before it can be broadcast to other blockchain nodes. In other words, if data shared by a blockchain node is directly sent to other blockchain nodes without encryption, then all other blockchain nodes acquire the data shared by the blockchain node, and it won't be possible to perform permission control, which will be further described below.

[53] FIG. 2 is a flow chart of a data encryption method according to some embodiments of the present description, comprising the following steps:

[54] S200: receiving, by an encryption device, data sent by a blockchain node.

[55] S202: encrypting a part of the data.

[56] S204: returning the encrypted data to the blockchain node.

[57] In some embodiments of the present description, the encryption device receives data sent by a blockchain node and then only encrypts a part of the data. This is because, in addition to data that has the sharing value, the data shared by a blockchain node typically further comprises some explanatory information (e.g., data categories, sources, summaries, and the like). For example, with regard to the data of a blacklist of clients in arrears with rent shared by a housing rental company, client names, contact information, and ID numbers in the blacklist are the data with value, while the name and institution code of the housing rental company are explanatory information.

[58] Therefore, there are some data in the data shared by a blockchain node that have no need to be confidential to other blockchain nodes. Therefore, the encryption device just needs to encrypt a part of the received data. Moreover, encrypting a part of the data facilitates data query by blockchain nodes, which will be described later. For the purpose of easy description, the encrypted part of each encrypted data will be referred to as transaction data, and the unencrypted part of each encrypted data will be referred to as non-transaction data hereinafter.

[59] Upon receiving the encrypted data, the blockchain node can execute the above step S104, i.e., sending the encrypted data to other blockchain nodes associated with the blockchain for consensus verification and then cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[60] FIG. 3 is a flow chart of a data acquisition method according to some embodiments of the present description, comprising the following steps:

[61] S300: determining, by a blockchain node (e.g., one of the blockchain nodes associated with a blockchain), data to be queried in encrypted data stored in a blockchain.

[62] The flow shown in FIG. 3 is a flow in which a blockchain node queries the data in a blockchain, and data queried by a blockchain node is often stored by other blockchain nodes in the blockchain in the flow shown in FIG. 1.

[63] In some embodiments of the present description, if the encrypted data stored in the blockchain is partially encrypted (the encrypted data can be obtained through the flow shown in FIG. 2), then a blockchain node that wants to query the data can select data that it wants to query (referred to as data to be queried herein) according to the unencrypted part of each encrypted data (non-transaction data) stored in the blockchain. Apparently, the transaction data in the data to be queried are encrypted, and the data that the blockchain node wants to query will be obtained only after encryption by an encryption device.

[64] In addition, the blockchain node can also determine a data query condition; determine, through a smart contract and according to the data query condition, whether the unencrypted part comprised in each encrypted data stored in the blockchain satisfies the data query condition; if the data query condition is satisfied, use the encrypted data as the encrypted data to be queried; if the data query condition is not satisfied, not use the encrypted data as the encrypted data to be queried. In some embodiments, the data query condition can be a keyword comprised in data to be queried as determined by the blockchain node, data source, the field to which the data belongs, and the like. For example, if a bank wants to acquire a client's house purchase records, the bank can conduct search with a keyword "house purchase record," which can retrieve non-transaction data that contains the above keyword, and the encrypted data comprising the retrieved non-transaction data is the data to be queried.

[65] It is known in the field of blockchain technologies that a smart contract may be essentially code that can be triggered and executed. It is typically preset in a blockchain and it is difficult to tamper with a smart contract stored in a blockchain. Therefore, the smart

contract can be used to execute a preset code logic, and the execution process is not interfered by blockchain nodes. In some embodiments of the present description, a smart contract is used in a plurality of different steps to execute different code logics, wherein the different code logics can be executed by different smart contracts or by the same smart contract, which is not limited herein. In summary, the role of a smart contract is to ensure that blockchains executes preset code logics and prevent blockchains from doing evil.

[66] S302: determining, through a smart contract, whether the blockchain node has a permission to decrypt the data to be queried; if the blockchain node has the permission, proceeding to the step S304; if the blockchain node does not have the permission, proceeding to the step S306.

[67] In some embodiments of the present description, attribute information of the data to be queried can be determined through a smart contract according to non-transaction data comprised in the data to be queried, and then it is determined whether attribute information that the blockchain node has a permission to query comprises attribute information of the data to be queried; if yes, the blockchain node has a permission to decrypt the data to be queried; otherwise, the blockchain node does not have a permission to decrypt the data to be queried.

[68] In some embodiments, attribute information of the data to be queried can be information such as source of the data to be queried, data category, and the like, all of which can be determined according to non-transaction data comprised in the data to be queried. The attribute information that the blockchain node has a permission to query is attribute information of preset data that the blockchain node has a permission to query.

[69] S304: sending the data to be queried to an encryption device through the smart contract.

[70] If the blockchain node has a permission to decrypt the data to be queried, the blockchain node can send the data to be queried to an encryption device through the smart contract to cause the encryption device to decrypt the data to be queried and return data obtained from the decryption to the blockchain node. As such, the blockchain node obtains the plaintext data that it desires.

[71] S306: refusing to send the data to be queried to the encryption device.

[72] According to the data storage method shown in FIG. 1, the data encryption method shown in FIG. 2, and the data query method shown in FIG. 3, it can be seen that, first, data shared by blockchain nodes are stored in a distributed manner in a blockchain respectively maintained by the blockchain nodes, and consequently it is difficult to tamper with the data stored in the blockchain, leading to high security; second, the data are encrypted before being

stored in the blockchain, and if a blockchain node wants to acquire data, its permission needs to be verified via a smart contract, and encrypted data that the blockchain node has a permission to acquire will be sent to an encryption device for decryption and ultimately returned to the blockchain node; third, each blockchain node can rapidly query data from the blockchain maintained by itself, which improves the efficiency of data query.

[73] Moreover, to encourage institutions to share data, a data trade mechanism can be introduced into the flows shown in FIG. 1 and FIG. 3.

[74] In one example, when a blockchain node shares data in the flow shown in FIG. 1, the blockchain node can determine a first value corresponding to the data through a smart contract, and then send the blockchain node's node identifier and the first value to other blockchain nodes to cause the node identifier and the first value to be stored in the blockchain after a consensus verification on the node identifier and the first value by the blockchain nodes passed.

[75] In some embodiments, the first value can be used to represent a value added into the account of the blockchain node. In other words, the first value is actually an award obtained by the blockchain node for sharing the data. The first value should be determined through a smart contract to prevent a blockchain node from fabricating the first value without authorization. Different data can correspond to different first values, which can be pre-determined in the smart contract according to the needs.

[76] It should be noted that, in the flow shown in FIG. 1, a blockchain node can also determine the first value before sending the encrypted data to other blockchain nodes for consensus verification, and then broadcast its own node identifier, the first value, and the encrypted data together to other blockchain nodes for consensus verification. In summary, in addition to the shared data, a node identifier of the blockchain node that shares data and the first value can also be stored in a blockchain as evidence and for publication if they pass consensus verification, which is equivalent to a balance increase of the account of the blockchain node. The blockchain node can use the balance of its account for consumption to exchange for data shared by other blockchain nodes. As such, the blockchain nodes in a blockchain network form a trade mechanism that encourages data sharing.

[77] Correspondingly, when a blockchain node wants to acquire data shared by other blockchain nodes in the flow shown in FIG. 3, the blockchain node can determine, through a smart contract, a second value corresponding to the data; send the blockchain node's node identifier and the second value to other blockchain nodes to cause each of the other

blockchain nodes to store the node identifier and the second value in the blockchain after performing consensus verification on the node identifier and the second value with success.

[78] In some embodiments, the second value can be used to represent a value deducted from the account of the blockchain node. In other words, the second value can be regarded as expenses for querying the data by the blockchain node. The second value corresponding to the data should also be determined through a smart contract. The expenses made by the blockchain node can be deducted from the balance of its account. The blockchain node can broadcast its own node identifier and the second value to other blockchain nodes for consensus verification. If the consensus verification is passed, the node identifier of the blockchain node that acquires the data and the second value will also be stored in a blockchain as evidence and for publication.

[79] Therefore, a blockchain not only stores data shared by institutions, but also publishes records of awards or expenses of the institutions. By checking the transaction records on the blockchain, an account balance can be determined for each institution.

[80] Furthermore, the blockchain node that acquires the data can determine, through a smart contract, a third value corresponding to the data and node identifiers of other blockchain nodes sharing the data; then send the determined node identifiers and the third value to other blockchain nodes to cause each of the other blockchain nodes to store the determined node identifiers and the third value in the blockchain after performing consensus verification on the determined node identifiers and the third value with success.

[81] In some embodiments, the third value can be used to represent a value added into the accounts of other blockchain nodes sharing the data. In other words, the third value can be regarded as an award for a blockchain node because the data shared by the blockchain node is queried.

[82] In some embodiments of the present description, moreover, the blockchain node can send a refill request to a refill device to cause the refill device to determine a refill certificate according to the refill request and return the refill certificate to the blockchain node; the refill certificate comprises a fourth value and the fourth value can be used to represent a value added into the account of the blockchain node; the blockchain node can send its own node identifier and the refill certificate to other blockchain nodes to cause each of the other blockchain nodes to store the node identifier and the fourth value in the blockchain after performing consensus verification on the node identifier and the refill certificate with success.

[83] In some embodiments, the blockchain node can send the fourth value to other blockchain nodes to cause each of the other blockchain nodes to compare the received fourth value and the fourth value contained in the refill certificate; if they are consistent, the verification is passed. It is also feasible that the blockchain node does not send the fourth value to other blockchain nodes, and the blockchain nodes can directly store the fourth value contained in the refill certificate in the blockchain.

[84] To be able to continue acquiring data when a blockchain node does not have enough balance in its account, the blockchain node can refill its account, and the refill result can also be stored in a blockchain as evidence after passing consensus verification performed by the blockchain nodes.

[85] In some embodiments, the refill device is not a blockchain node, but is a device responsible for processing and settling a refill request from a blockchain node and issuing a refill certificate.

[86] FIG. 4 is a schematic diagram of an architecture of a data storage and query system according to some embodiments of the present description. As shown in FIG. 4, servers of institutions A to C act as blockchain nodes and form a blockchain network. The server of each institution maintains a blockchain, and the server of each institution can share data into the blockchain or query data from the blockchain. In the architecture shown in FIG. 4, moreover, there are a third-party encryption device and a refill device. The encryption device is configured to encrypt and decrypt data to prevent an institution from decrypting data in the blockchain without authorization; the refill device is configured to issue a refill certificate to the server of an institution that purchases the fourth value.

[87] According to the data storage method shown in FIG. 1, some embodiments of the present description correspondingly provide a data storage apparatus, as shown in FIG. 5, comprising:

[88] a first sending module 501 configured to send data to an encryption device to cause the encryption device to encrypt the received data and return the encrypted data to the blockchain node;

[89] a receiving module 502 configured to receive the encrypted data returned by the encryption device;

[90] a second sending module 503 configured to send the encrypted data to other blockchain nodes to cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[91] The apparatus further comprises: a processing module 504 configured to determine a first value corresponding to the data through a smart contract, the first value being used to represent a value added into the account of the apparatus; send its own node identifier and the first value to other blockchain nodes to cause the node identifier and the first value to be stored in the blockchain after a consensus verification on the node identifier and the first value by the blockchain nodes passed.

[92] The various modules and units of the data storage apparatus may be implemented as software instructions. That is, the data storage apparatus may comprise a processor and a non-transitory computer-readable storage medium storing instructions that, when executed by the processor, cause the data storage apparatus to perform various steps and methods of the modules and units described above. In some embodiments, the data storage apparatus may include a computer, a server, a mobile phone, a pad, etc. In some embodiments, the data storage apparatus may act as a blockchain node.

[93] According to the data encryption method shown in FIG. 2, some embodiments of the present description correspondingly provide a data encryption apparatus, as shown in FIG. 6, comprising:

[94] a receiving module 601 configured to receive data sent by a blockchain node;

[95] an encrypting module 602 configured to encrypt a part of the data;

[96] a returning module 603 configured to return the encrypted data to the blockchain node to cause the blockchain node to send the encrypted data to other blockchain nodes for consensus verification and then cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[97] The various modules and units of the data encryption apparatus may be implemented as software instructions. That is, the data encryption apparatus may comprise a processor and a non-transitory computer-readable storage medium storing instructions that, when executed by the processor, cause the data encryption apparatus to perform various steps and methods of the modules and units described above. In some embodiments, the data encryption apparatus may include a computer, a server, a mobile phone, a pad, etc.

[98] According to the data query method shown in FIG. 3, some embodiments of the present description correspondingly provide a data query apparatus, as shown in FIG. 7, comprising:

[99] a determining module 701 configured to determine data to be queried in encrypted data stored in a blockchain;

[100] a determining and processing module 702 configured to determine, through a smart

contract, whether the apparatus has a permission to decrypt the data to be queried; if the blockchain node has the permission, send the data to be queried to an encryption device through the smart contract to cause the encryption device to decrypt the data to be queried and return data obtained from the decryption to the apparatus; if the blockchain node does not have the permission, refuse to send the data to be queried to the encryption device.

[101] Each encrypted data stored in the blockchain comprises an unencrypted part.

[102] The determining module 701 is configured to determine a data query condition; determine, through a smart contract and according to the data query condition, whether the unencrypted part comprised in each encrypted data stored in the blockchain satisfies the data query condition; if the data query condition is satisfied, use the encrypted data as data to be queried; if the data query condition is not satisfied, not use the encrypted data as data to be queried.

[103] The apparatus further comprises: a first processing module 703 configured to, when the blockchain node has a permission to decrypt the data to be queried, determine, through a smart contract, a second value corresponding to the data returned by the encryption device, the second value being used to represent a value deducted from the account of the blockchain node; send, through the smart contract, the blockchain node's node identifier and the second value to other blockchain nodes to cause each of the other blockchain nodes to store the node identifier and the second value in the blockchain after performing consensus verification on the node identifier and the second value with success.

[104] The apparatus further comprises: a second processing module 704 configured to, when the blockchain node has a permission to decrypt the data to be queried, determine, through a smart contract, a third value corresponding to the data and node identifiers of other blockchain nodes sharing the data, the third value being used to represent a value added into the accounts of other blockchain nodes sharing the data; send, through the smart contract, the determined node identifiers and the third value to other blockchain nodes to cause each of the other blockchain nodes to store the determined node identifiers and the third value in the blockchain after performing consensus verification on the determined node identifiers and the third value with success.

[105] The apparatus further comprises: a third processing module 705 configured to send a refill request to a refill device to cause the refill device to determine a refill certificate according to the refill request and return the refill certificate to the blockchain node; the refill certificate comprising a fourth value and the fourth value being used to represent a value

added into the account of the blockchain node; send its own node identifier and the refill certificate to other blockchain nodes to cause each of the other blockchain nodes to store the node identifier and the fourth value in the blockchain after performing consensus verification on the node identifier and the refill certificate with success.

[106] The various modules and units of the data query apparatus may be implemented as software instructions. That is, the data query apparatus may comprise a processor and a non-transitory computer-readable storage medium storing instructions that, when executed by the processor, cause the data query apparatus to perform various steps and methods of the modules and units described above. In some embodiments, the data query apparatus may include a computer, a server, a mobile phone, a pad, etc. In some embodiments, the data storage apparatus may act as a blockchain node.

[107] According to the data storage method shown in FIG. 1, some embodiments of the present description correspondingly provide a data storage device, as shown in FIG. 8, comprising one or more processors and memories, wherein the memories store programs, and being configured to execute the following steps by the one or more processors:

[108] sending data to an encryption device to cause the encryption device to encrypt the received data and return the encrypted data to the blockchain node;

[109] receiving the encrypted data returned by the encryption device;

[110] sending the encrypted data to other blockchain nodes to cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[111] According to the data storage method shown in FIG. 1, some embodiments of the present description correspondingly provide a data encryption device, as shown in FIG. 9, comprising one or more processors and memories, wherein the memories store programs, and being configured to execute the following steps by the one or more processors:

[112] receiving data sent by a blockchain node;

[113] encrypting a part of the data;

[114] returning the encrypted data to the blockchain node to cause the blockchain node to send the encrypted data to other blockchain nodes for consensus verification and then cause each of the other blockchain nodes to store the encrypted data in the blockchain after performing consensus verification on the encrypted data with success.

[115] According to the data query method shown in FIG. 2, some embodiments of the present description correspondingly provide a data query device, as shown in FIG. 10, comprising

one or more processors and memories, wherein the memories store programs, and being configured to execute the following steps by the one or more processors:

[116] determining data to be queried in encrypted data stored in a blockchain;

[117] determining, through a smart contract, whether the blockchain node has a permission to decrypt the data to be queried;

[118] if the blockchain node has the permission, sending the data to be queried to an encryption device through the smart contract to cause the encryption device to decrypt the data to be queried and return data obtained from the decryption to the blockchain node;

[119] if the blockchain node does not have the permission, refusing to send the data to be queried to the encryption device.

[120] The embodiments in the present description are described in a progressive manner with each embodiment focused on differences from other embodiments, and the embodiments may be mutually referenced for identical or similar parts thereof. For the devices shown in FIGS. 8-10, in particular, the description thereof is relatively simple as they are substantially similar to the method embodiments. The description of the method embodiments may be referenced for related parts thereof.

[121] In the 1990s, an improvement to a technology can be differentiated into a hardware improvement (e.g., an improvement to a circuit structure, such as a diode, a transistor, a switch, and the like) or a software improvement (an improvement to a flow of a method). Along with the technological development, however, many current improvements to method flows can be deemed as direct improvements to hardware circuit structures. Designers almost always obtain a corresponding hardware circuit structure by programming an improved method flow into a hardware circuit. Therefore, it is not that an improvement to a method flow cannot be realized with a hardware module. For example, Programmable Logic Device (PLD) (e.g., Field Programmable Gate Array (FPGA)) is such an integrated circuit that its logic functions are determined by a user through programming the device. A designer programs on his/her own to “integrate” a digital system onto one piece of PLD, who does not need to ask a chip manufacturer to design and manufacture a dedicated IC chip. At present, moreover, this type of programming has mostly been implemented through “logic compiler” software, rather than manually manufacturing the IC chips. The logic compiler software is similar to a software compiler used for program development and writing, while a particular programming language must be used for writing source codes prior to compiling, which is referred to as a Hardware Description Language (HDL). There is not just one, but many types

of HDL, such as ABEL (Advanced Boolean Expression Language), AHDL (Altera Hardware Description Language), Confluence, CUPL (Cornell University Programming Language), HDCal, JHDL (Java Hardware Description Language), Lava, Lola, MyHDL, PALASM, and RHDL (Ruby Hardware Description Language). What are most commonly used right now include VHDL (Very-High-Speed Integrated Circuit Hardware Description Language) and Verilog. A person skilled in the art should also be aware that it would be very easy to obtain a hardware circuit to implement a logic method flow by using the above HDLs to carry out a little bit logic programming on the method flow and program it into an IC.

[122] A controller may be implemented in any proper manner. For example, a controller may be, for example, in a form of a microprocessor or processor, as well as a computer readable medium that stores computer readable program codes (e.g., software or firmware) capable of being executed by the (micro)processor, a logic gate, a switch, an Application Specific Integrated Circuit (ASIC), a programmable logic controller, and an embedded microcontroller. Examples of the controller include, but are not limited to, the following microcontrollers: ARC 625D, Atmel AT91SAM, Microchip PIC18F26K20, and Silicone Labs C8051F320. A memory controller may further be implemented as a part of a control logic of a memory. A person skilled in the art should also be aware that, in addition to that a controller is implemented in a manner of pure computer readable program codes, it is totally feasible to perform logic programming on steps of a method to enable a controller to implement the same functions in a form of a logic gate, a switch, an ASIC, a programmable logic controller and an embedded microcontroller. Therefore, such a controller can be deemed as a hardware part, while apparatuses comprised therein and configured to carry out various functions may also be deemed as a structure inside the hardware part. Alternatively, apparatuses configured to carry out various functions may even be deemed as both software modules to implement a method and a structure inside a hardware part.

[123] The system, apparatus, module, or unit described in the above embodiments may be implemented, for example, by a computer chip or entity or implemented by a product having a function. A typical implementation device is a computer. In one example, a computer may be, for example, a personal computer, a laptop computer, a cellular phone, a camera phone, a smart phone, a personal digital assistant, a medium player, a navigation device, an email device, a game console, a tablet computer, a wearable device, or a combination of any devices in these devices.

[124] For the convenience of description, the above apparatus is divided into various units

according to functions for description. Functions of the units may be implemented in one or multiple pieces of software and/or hardware when implementing the present description.

[125] A person skilled in the art should understand that the embodiments of the present invention may be provided as a method, a system, or a computer program product. Therefore, the present invention may be implemented as a complete hardware embodiment, a complete software embodiment, or an embodiment combing software and hardware. Moreover, the present invention may be in the form of a computer program product implemented on one or more computer usable storage media (including, but not limited to, a magnetic disk memory, CD-ROM, an optical memory, and the like) comprising computer usable program codes therein.

[126] The present invention is described with reference to flow charts and/or block diagrams of the method, device (system) and computer program product according to the embodiments of the present invention. A computer program instruction may be used to implement each process and/or block in the flow charts and/or block diagrams and a combination of processes and/or blocks in the flow charts and/or block diagrams. These computer program instructions may be provided for a general-purpose computer, a special-purpose computer, an embedded processor, or a processor of other programmable data processing devices to generate a machine, so that the instructions executed by a computer or a processor of other programmable data processing devices generate an apparatus for implementing a specified function in one or more processes in the flow charts and/or in one or more blocks in the block diagrams.

[127] These computer program instructions may also be stored in a computer readable memory that can instruct a computer or other programmable data processing devices to work in a particular manner, such that the instructions stored in the computer readable memory generate a manufactured article that includes an instruction apparatus. The instruction apparatus implements a specified function in one or more processes in the flow charts and/or in one or more blocks in the block diagrams.

[128] These computer program instructions may also be loaded onto a computer or other programmable data processing devices to cause a series of operational steps to be executed on the computer or other programmable devices to generate computer-implemented processing. Therefore, the instructions executed on the computer or other programmable devices provide steps for implementing a specified function in one or more processes in the flow charts and/or in one or more blocks in the block diagrams.

[129] In a typical configuration, the computation device includes one or more processors (CPUs), input/output interfaces, network interfaces, and a memory.

[130] The memory may include computer readable media, such as a volatile memory, a Random Access Memory (RAM), and/or a non-volatile memory, e.g., a Read-Only Memory (ROM) or a flash RAM. The memory is an example of a computer readable medium.

[131] Computer readable media include permanent, volatile, mobile and immobile media, which can implement information storage through any method or technology. The information may be computer readable instructions, data structures, program modules or other data. Examples of storage media of computers include, but are not limited to, Phase-change RAMs (PRAMs), Static RAMs (SRAMs), Dynamic RAMs (DRAMs), other types of Random Access Memories (RAMs), Read-Only Memories (ROMs), Electrically Erasable Programmable Read-Only Memories (EEPROMs), flash memories or other memory technologies, Compact Disk Read-Only Memories (CD-ROMs), Digital Versatile Discs (DVDs), or other optical memories, cassettes, cassette, and disk memories or other magnetic memory devices or any other non-transmission media, which can be used for storing information accessible to a computation device. According to the definitions herein, the computer readable media do not include transitory media, such as modulated data signals and carriers.

[132] It should be further noted that the terms of “including”, “comprising” or any other variants thereof intend to encompass a non-exclusive inclusion, such that a process, method, commodity or device comprising a series of elements not only comprises these elements, but also comprises other elements that are not specifically listed, or further comprises elements that are inherent to the process, method, commodity or device. When there is no further restriction, elements defined by the statement “comprising one…” does not exclude that a process, method, commodity or device comprising the above elements further comprises additional identical elements.

[133] A person skilled in the art should understand that the embodiments of the present description may be provided as a method, a system, or a computer program product. Therefore, the present description may be implemented as a complete hardware embodiment, a complete software embodiment, or an embodiment combing software and hardware. Moreover, the present description may be in the form of a computer program product implemented on one or more computer usable storage media (including, but not limited to, a magnetic disk memory, CD-ROM, an optical memory, and the like) including computer

usable program codes.

[134] The present description may be described in a regular context of a computer executable instruction that is executed by a computer, such as a program module. Generally, the program module comprises a routine, a program, an object, a component, a data structure, and the like for executing a particular task or implementing a particular abstract data type. The present description may also be practiced in distributed computing environments. In these distributed computing environments, remote processing devices connected via communication networks carry out tasks. In the distributed computing environments, a program module can be located in local and remote computer storage media, including storage devices.

[135] Only embodiments of the present description are described above, which are not used to limit the present description. To a person skilled in the art, the present description may have various modifications and changes. Any modification, equivalent substitution or improvement made within the spirit and principle of the present description shall be encompassed by the claims of the present description.

Claims

1. A method, comprising:

receiving, from a first blockchain node of a blockchain network, a query for encrypted data stored in a blockchain, wherein the encrypted data comprises an unencrypted part;

determining, through one or more smart contracts and based on the unencrypted part, first attribute information of the encrypted data;

determining whether the first blockchain node has permission to query data associated with the first attribute information;

in response to determining that the first blockchain node has the permission:

sending the encrypted data to a device to decrypt the encrypted data to obtain decrypted data and return the decrypted data to the first blockchain node, wherein the device is a third-party device that is not a blockchain node;

after the decrypted data is returned to the first blockchain node, determining, through the one or more smart contracts, an expense to be deducted from an account of the first blockchain node for querying the encrypted data; and

sending a node identifier of the first blockchain node and the expense to other blockchain nodes of the blockchain network, enabling each of the other blockchain nodes to store the node identifier and the expense in the blockchain after the node identifier and the expense are successfully verified through at least a first consensus verification.

2. The method of claim 1, further comprising:

determining, through the one or more smart contracts, a second node identifier of a second blockchain node of the blockchain network sharing the encrypted data, and an award to be added into a second account of the second blockchain node for sharing the encrypted data; and

sending the second node identifier and the award to the other blockchain nodes, enabling each of the other blockchain nodes to store the second node identifier and the award in the blockchain after the second node identifier and the award are successfully verified through at least a second consensus verification.

3. The method of claim 1, further comprising:

sending, by the first blockchain node, a refill request to a refill device, enabling the refill device to determine a refill certificate according to the refill request and return the refill certificate to the first blockchain node, wherein the refill certificate contains a value that is added into the account of the first blockchain node; and

sending the node identifier and the refill certificate to the other blockchain nodes, enabling each of the other blockchain nodes to store the node identifier and the value in the blockchain after at least a second consensus verification on the node identifier and the refill certificate is successfully performed.

4. The method of claim 3, further comprising: sending the value to the other blockchain nodes, enabling each of the other blockchain nodes to compare the sent value and the value contained in the refill certificate during the at least second consensus verification.

5. The method of claim 1, further comprising:

sending, by a second blockchain node of the blockchain network, unencrypted data to the device, enabling the device to encrypt the unencrypted data and return the encrypted data to the second blockchain node;

determining, through the one or more smart contracts, an award to be added into a second account of the second blockchain node for sharing the encrypted data; and

sending the encrypted data, a second node identifier of the second blockchain node, and the award to the blockchain network to store the encrypted data, the second node identifier, and the award in the blockchain after the encrypted data, the second node identifier, and the award are successfully verified through at least a second consensus verification.

6. A system, comprising:

one or more processors; and

one or more non-transitory computer-readable storage medium storing instructions that, when executed by the one or more processors, cause the one or more processors to perform operations comprising:

receiving, from a first blockchain node of a blockchain network, a query for encrypted data stored in a blockchain, wherein the encrypted data comprises an unencrypted part;

determining, through one or more smart contracts and based on the unencrypted part, first attribute information of the encrypted data;

determining whether the first blockchain node has permission to query data associated with the first attribute information;

in response to determining that the first blockchain node has the permission:

sending the encrypted data to a device to decrypt the encrypted data to obtain decrypted data and return the decrypted data to the first blockchain node, wherein the device is a third-party device that is not a blockchain node;

after the decrypted data is returned to the first blockchain node, determining, through the one or more smart contracts, an expense to be deducted from an account of the first blockchain node for querying the encrypted data; and

sending a node identifier of the first blockchain node and the expense to other blockchain nodes of the blockchain network, enabling each of the other blockchain nodes to store the node identifier and the expense in the blockchain after the node identifier and the expense are successfully verified through at least a first consensus verification.

7. The system of claim 6, wherein the operations further comprise:

determining, through the one or more smart contracts, a second node identifier of a second blockchain node of the blockchain network sharing the encrypted data, and an award to be added into a second account of the second blockchain node for sharing the encrypted data; and

sending the second node identifier and the award to the other blockchain nodes, enabling each of the other blockchain nodes to store the second node identifier and the award in the blockchain after the second node identifier and the award are successfully verified through at least a second consensus verification.

8. The system of claim 6, wherein the operations further comprise:

sending, by the first blockchain node, a refill request to a refill device, enabling the refill device to determine a refill certificate according to the refill request and return the refill certificate to the first blockchain node, wherein the refill certificate contains a value that is added into the account of the first blockchain node; and

sending the node identifier and the refill certificate to the other blockchain nodes, enabling each of the other blockchain nodes to store the node identifier and the value in the blockchain after at least a second consensus verification on the node identifier and the refill certificate is successfully performed.

9. The system of claim 8, wherein the operations further comprise: sending the value to the other blockchain nodes, enabling each of the other blockchain nodes to compare the sent value and the value contained in the refill certificate during the at least second consensus verification.

10. The system of claim 6, wherein the operations further comprise:

sending, by a second blockchain node of the blockchain network, unencrypted data to the device, enabling the device to encrypt the unencrypted data and return the encrypted data to the second blockchain node;

determining, through the one or smart contracts, an award to be added into a second account of the second blockchain node for sharing the encrypted data; and

sending the encrypted data, a second node identifier of the second blockchain node, and the award to the blockchain network to store the encrypted data, the second node identifier, and the award in the blockchain after the encrypted data, the second node identifier, and the award are successfully verified through at least a second consensus verification.

11. A non-transitory computer-readable storage medium storing instructions that, when executed by a processor, cause the processor to perform operations comprising:

receiving, from a first blockchain node of a blockchain network, a query for encrypted data stored in a blockchain, wherein the encrypted data comprises an unencrypted part;

determining, through one or more smart contracts and based on the unencrypted part, first attribute information of the encrypted data;

determining whether the first blockchain node has permission to query data associated with the first attribute information;

in response to determining that the first blockchain node has the permission:

sending the encrypted data to a device to decrypt the encrypted data to obtain decrypted data and return the decrypted data to the first blockchain node, wherein the device is a third-party device that is not a blockchain node;

after the decrypted data is returned to the first blockchain node, determining, through the one or more smart contracts, an expense to be deducted from an account of the first blockchain node for querying the encrypted data; and

sending a node identifier of the first blockchain node and the expense to other blockchain nodes of the blockchain network, enabling each of the other blockchain nodes to store the node identifier and the expense in the blockchain after the node identifier and the expense are successfully verified through at least a first consensus verification.

12. The non-transitory computer-readable storage medium of claim 11, wherein the operations further comprise:

determining, through the one or more smart contracts, a second node identifier of a second blockchain node of the blockchain network sharing the encrypted data, and an award to be added into a second account of the second blockchain node for sharing the encrypted data; and

sending the second node identifier and the award to the other blockchain nodes, enabling each of the other blockchain nodes to store the second node identifier and the award in the blockchain after the second node identifier and the award are successfully verified through at least a second consensus verification.

13. The non-transitory computer-readable storage medium of claim 11, wherein the operations further comprise:

sending, by the first blockchain node, a refill request to a refill device, enabling the refill device to determine a refill certificate according to the refill request and return the refill certificate to the first blockchain node, wherein the refill certificate contains a value that is added into the account of the first blockchain node; and

sending the node identifier and the refill certificate to the other blockchain nodes, to cause enabling each of the other blockchain nodes to store the node identifier and the value in the blockchain after at least a second consensus verification on the node identifier and the refill certificate is successfully performed.

14. The non-transitory computer-readable storage medium of claim 13, wherein the operations further comprise: sending the value to the other blockchain nodes, enabling each of

the other blockchain nodes to compare the sent value and the value contained in the refill certificate during the at least second consensus verification.

15. The method of claim 1, wherein the first blockchain node has a permission to query preset data associated with second attribute information; and determining whether the first block chain node has permission to query data associated with the first attribute information comprises: determining whether the second attribute information comprises the first attribute information.

16. The system of claim 6, wherein the first blockchain node has a permission to query preset data associated with second attribute information; and determining whether the first block chain node has permission to query data associated with the first attribute information comprises: determining whether the second attribute information comprises the first attribute information.

17. The non-transitory computer-readable storage medium of claim 11, wherein the first blockchain node has a permission to query preset data associated with second attribute information; and determining whether the first block chain node has permission to query data associated with the first attribute information comprises: determining whether the second attribute information comprises the first attribute information.