

NETWORK TRANSACTION VERIFICATION METHOD BASED ON A PLURALITY OF NODES, SYSTEM THEREFOR, AND STORAGE MEDIUM

TECHNICAL FIELD OF THE INVENTION

5 The present disclosure relates to the field of electronic commerce, and more particularly, to a network transaction verification method based on a plurality of nodes, a computer system, and a storage medium thereof.

BACKGROUND

10 With the assistance of the Internet technology and basic communication device, electronic commerce provides a lot of convenience for performing transaction between a merchant and a user, parties in a transaction may only directly reach a transaction agreement and transfer funds through a network. Therefore, unlike traditional transaction methods, parties in electronic transaction cannot directly confirm the identity of each other and whether goods/services and funds are delivered in a
15 way specified in the agreement during the transaction. As such, parties in the transaction often need to verify the identity of each other through a third-party transaction platform and guarantee the transaction.

 When using a third-party transaction platform to perform a transaction, the merchant or the user needs to provide some proof of identity first. At the same time, the merchant or the user may also
20 need to provide additional guarantee funds. The third-party transaction platform needs to spend corresponding costs to verify identities of parties in the transaction and also deal with possible disputes in the transaction. Especially for large volume of daily micro-transactions, a background server of the third-party transaction platform needs to verify identities of both parties in each transaction with high concurrency to confirm contents of the transaction, and to ensure a security of
25 funds, thereby guaranteeing that the transaction is genuinely intended by both parties with corresponding accounts in the platform, as such the third-party platform is required to keep high operational and maintenance capability and cost.

 For transaction of large commodity, especially for an initial transaction between two parties, both parties in the transaction generally use a traditional off-line transaction method at present to
30 ensure fairness for both parties.

 In addition, since related public blockchain technologies (such as Bitcoin, Ethereum and other

cryptocurrencies) do not have any verification mechanism and do not provide data confidentiality, these non-verification technologies require additional resources to prevent malicious behaviors, thus transaction processing capability is being affected. Currently, a transaction processing capacity of a Bitcoin network is about 3.3 and 7 transactions per second, and a transaction processing capacity of the Ethereum is about 25 transactions per second.

Therefore, it is a need to provide a transaction method of high processing efficiency with data confidentiality to deal with massive amount of transactions.

SUMMARY OF THE INVENTION

To address the above technical problem, the present application provides a network transaction verification method based on a plurality of nodes, a corresponding computer system, and a storage medium, so as to alleviate problems in related technical solutions. The present application provides the technical solution to efficiently verify validity in each transaction.

In order to achieve the above objective, the present application provides the following technical solutions.

Firstly, the present application provides a network transaction verification method based on a plurality of nodes, comprises the following steps: identifying authenticated identities of a plurality of nodes participating in the network transaction; selecting a set of nodes from the plurality of nodes as verification nodes for verifying nodes in the network transaction; initializing a public P2P channel transmission data between the plurality of nodes and a current public transaction record linked list for recording the network transaction; and generating and adding, by means of an IBFT (Istanbul Byzantine Fault Tolerant) consensus, a data block containing transaction information and signature to the current public transaction record linked list. The nodes may participate in a client-end in the network transaction. The client-end stores identity information which have been authenticated. The authenticated identity information may be stored on a device of the client.

In the above embodiment of the present application, when a number of the verification nodes is $3F+1$, the step of initializing the public P2P channel transmission data between the plurality of nodes and the current public transaction record linked list for recording the network transaction further includes the following sub-steps of: collecting verification results returned by at least $2F+1$ number of verification nodes; and counting the collected verification results, and when the data block has verified by greater than or equal to $F+1$ number of the verification nodes, recording the data block to end of the current public transaction record linked list.

In one or more embodiments of the present application, data block is write-locked during process

of being verified, so that the data block is being processed by all the nodes in a read-only manner.

In one or more embodiments of the present application, data block is generated at a constant rate, and when there is no new transaction information, transaction information in the data block is empty.

5 In one or more embodiments of the present application, when the number of data blocks added to the current public transaction record linked list is greater than a updating threshold, the following step is also executed: when a verification node fails to pass verification by 2/3 of all the verification nodes, said verification nodes is being deselected from being one of the set of verification nodes.

10 Similarly, in one or more embodiments of the present application, when the number of data blocks added to the current public transaction record linked list is greater than the updating threshold, the following step is also executed: when a newly added node passes verification by 2/3 of all the verification nodes, selecting the newly added node as a new verification node.

In one or more embodiments of the present application, the updating threshold is 30,000 (which means that when every 30,000 data blocks are added, the above verification of node is executed).

15 In one or more embodiments of the present application, the present invention further comprises the following steps executed between any two or more nodes: initializing a private transaction transmission channel and a private transaction data linked list based on the current public transaction record linked list, and sharing a private transaction data link to the two or more nodes through the private transaction transmission channel; by means of a Raft consensus, adding a data block
20 containing transaction information and signature between the two or more nodes to end of the private transaction data linked list; and through the private transaction transmission channel, updating the private transaction data linked list to the two or more nodes; wherein the private transaction transmission channel is separated from the public P2P channel.

25 Secondly, the present application also provides a computer system. The computer system includes a memory, a processor, and a computer program stored on the memory and executable on the processor. When the processor executes the program, the following steps are executed: identifying authenticated identities of a plurality of nodes participating in a network transaction; selecting a set of nodes from the plurality of nodes as verification nodes for verifying nodes in the network transaction; initializing a public P2P channel transmission data between the plurality of nodes and a
30 current public transaction record linked list for recording the network transaction; and generating and adding, by means of an IBFT consensus, a data block containing transaction information and signature to the current public transaction record linked list.

In the embodiment of the present application, when a number of the verification nodes is $3F+1$,

the step of initializing the public P2P channel transmission data between the plurality of nodes and the current public transaction record linked list for recording the network transaction further includes the following sub-steps of: collecting verification results returned by at least $2F+1$ number of verification nodes; and counting the collected verification results, and when the data block has
5 verified by greater than or equal to $F+1$ number of verification nodes, recording the data block to end of the current public transaction record linked list.

In one or more embodiments of the present disclosure, the data block is write-locked during the process of being verified, so that the data block is being processed by all the nodes in a read-only manner.

10 In one or more embodiments of the present disclosure, the data block is generated at a constant rate, and when there is no new transaction information exists at present, transaction information in the data block is empty.

In one or more embodiments of the present application, when the number of data blocks added to the current public transaction record linked list is greater than a updating threshold, the following
15 step is also executed: when a verification node fails to pass verification by $2/3$ of all the verification nodes, said verification node is being deselected from the set of verification nodes.

Similarly, in one or more embodiments of the present application, when the number of data blocks added to the current public transaction record linked list is greater than the updating threshold, the following step is also executed: when a newly added node passes verification by $2/3$ of all the
20 verification nodes, selecting the newly added node as a new verification node.

In one or more embodiments of the present application, the above updating threshold is 30,000.

In one or more embodiments of the present disclosure, when the processor executes the program, the following steps executed between any two or more nodes: initializing a private transaction transmission channel and a private transaction data linked list based on the current public transaction
25 record linked list, and sharing a private transaction data link to the two or more nodes through the private transaction transmission channel; by means of a Raft consensus, adding a data block containing the transaction information and signature between the two or more nodes to end of the private transaction data linked list; and through the private transaction transmission channel, updating the private transaction data linked list to the two or more nodes; wherein the private transaction
30 transmission channel is separated from the public P2P channel.

Finally, the present application discloses a storage medium, wherein the storage medium stores a computer program. When the computer processor is configured to execute the following steps: identifying authenticated identities of a plurality of nodes participating in a network transaction;

selecting a set of nodes from the plurality of nodes as verification nodes for verifying nodes in the network transaction; initializing a public P2P channel transmission data between the plurality of nodes and a current public transaction record linked list for recording the network transaction; and generating and adding, by means of an IBFT consensus, a data block containing transaction
5 information and signature to the current public transaction record linked list.

In the embodiment of the present application, when a number of the verification nodes is $3F+1$, the step of initializing the public P2P channel transmission data between the plurality of nodes and the current public transaction record linked list for recording the network transaction further includes the following sub-steps of: collecting verification results returned by at least $2F+1$ number of
10 verification nodes; and counting the collected verification results, and when the data block has verified by greater than or equal to $F+1$ number of verification nodes, recording the data block to end of the current public transaction record linked list.

In one or more embodiments of the present application, the data block is write-locked during the process of being verified, so that the data block is processed by all the nodes in a read-only manner.

15 In one or more embodiments of the present application, the data block is generated at a constant rate, and when there is no new transaction information exists at present, transaction information in the data block is empty.

In one or more embodiments of the present application, when the number of data blocks added to the current public transaction record linked list is greater than a updating threshold, the following
20 step is also executed: when a verification node fails to pass verification by $2/3$ of all the verification nodes, said verification node is being deselected from the set of verification nodes.

Similarly, in one or more embodiments of the present application, when the number of the data blocks added to the current public transaction record linked list is greater than the updating threshold, the following step is also executed: when a newly added node passes verification by $2/3$ of all the
25 verification nodes, selecting the node as a new added node as a verification node.

In one or more embodiments of the present application, the above updating threshold is 30,000.

In one or more embodiments of the present application, when the computer program is configured to run, the following steps are being executed between any two or more nodes: initializing a private transaction transmission channel and a private transaction data linked list based on the
30 current public transaction record linked list, and sharing a private transaction data link to the two or more nodes through the private transaction transmission channel; by means of a Raft consensus, adding a data block containing the transaction information and signature between the two or more nodes to end of the private transaction data linked list; and through the private transaction

transmission channel, updating the private transaction data linked list to the two or more nodes; wherein the private transaction transmission channel is separated from the public P2P channel.

The present invention has the following advantages: under high throughput of transaction records, in the event some nodes are down or are controlled by malicious participants that verify transactions arbitrarily, transactions can still be properly verified through verification of transactions by a set of randomly selected nodes. Since identities of all transaction participants are authenticated, the present invention can track actual participants of each transaction, thus ensuring reliability of the transaction. Since different consensus algorithms (IBFT and Raft) are used in public and private transactions, respectively, the present invention can avoid transaction forking from happening. On the contrary, a public blockchain transaction solution using a proof of work (PoW) often suffers from forking (even hardfork), leading to a waste of resources.

BRIEF DESCRIPTION OF DRAWINGS

FIG. 1 is a flow chart of a network transaction verification method based on a plurality of nodes according to an embodiment of the present application;

FIG. 2 a schematic diagram of a network formed by the plurality of nodes used in network transaction verification method in FIG. 1;

FIG. 3 is a schematic diagram of a data structure of a data block and a current public transaction record linked list;

FIG. 4 is a flow chart of a sub-method of the steps of initializing a public P2P channel transmission data between a plurality of nodes and the current public transaction record linked list for recording the network transaction in FIG. 1;

FIG. 5 is a flow chart of transaction verification between any two or more nodes according to an embodiment of the present application; and

FIG. 6 is a frame diagram of a network transaction verification system based on a plurality of nodes according to an embodiment of the present application.

DETAILED DESCRIPTION OF THE INVENTION

The concepts, specific structures and technical effects produced by the present invention will be clearly and completely described in conjunction with the embodiments and the accompanying drawings so as to sufficiently understand the objectives, solutions and effects of the present invention. It should be noted that, the embodiments in the application and the features in the embodiments may

be combined with each other without conflict. The same reference numbers are used in each part of the drawing to denote identical or similar parts.

At present, in a bulk commodity transaction, parties in the transaction often need to pay considerable costs to verify identities, qualification, historical transaction record, and other information of each parties to ensure reliability of transactions. Reliability of a party in a transaction is generally based on historical transaction records of the party and commercial creditability of the party evaluated by other participants in the historical transaction records. This information is used as references when parties evaluate each other's commercial creditability.

In the embodiment of the present application, this reference information is used as indexes for measuring security of a transaction, thus providing a security test for e-commerce transaction for the parties in the transaction. In applying the present invention, identities of two or more parties participating in a transaction may be verified on various terminals (such as a notebook computer, a desktop computer, a smart phone, a tablet computer, and the like) through a conventional identity verification method in the field that is associated with the terminals. The corresponding terminal then become one of the plurality of nodes for use in network transaction verification method. Specifically, FIG 1 depicts a flow chart of a network transaction verification method based on a plurality of nodes according to an embodiment of the present invention. The method includes the following steps of: identifying identities of a plurality of nodes participating in a network transaction; selecting a set of nodes from the plurality of nodes as verification nodes for verifying nodes in the network transaction; initializing a public P2P channel transmission data between the plurality of nodes and a current public transaction record linked list for recording the network transaction; and generating and adding, by means of an IBFT consensus, a data block containing transaction information and a signature to the current public transaction record linked list.

The IBFT consensus uses a proof of authority (PoA) based on selected verification nodes having identity authenticated as commercial credit evaluation of each participants in the transaction. The proof of authority uses a set of verification nodes to verify a new transaction, and through a consensus of most of the verification nodes, the new network transaction generates a corresponding data block, which is then added to the transaction record. According to the IBFT consensus, each verification node may be randomly selected and is dynamically changed in subsequent transaction verification processes. Whenever a new network transaction needs to be verified, one of all the verification nodes is randomly selected as "the Proposer". The randomly selected proposer generates a corresponding data block according to the new network transaction, which is submitted to all the verification nodes for verification. These verification nodes and other nodes form a network structure as shown in FIG. 2. According to one or more embodiments of the present invention, each data block is verified by a

different set of verification nodes; after a consensus is reached, a digital signature on the data block is recorded. The verification nodes verify each transaction and determine whether a data block can be added to the current public transaction record linked list; on the other hand, each of the ordinary nodes have a complete copy of the current public transaction record linked list such that the transaction
5 information and the verification results of each data blocks can be transmitted between each nodes, as such the nodes can quickly respond to a newly generated data block. After a data block passes the verification, it is added to the current public transaction record linked list, or in the case a data block fails to pass the verification and it is discarded, a new proposer may be randomly selected from all the verification nodes, or the same proposer is continuously being used, and the above process is
10 repeated to verify subsequent network transaction. When a data block fails to pass the verification, the proposer in this round of verification may be replaced, and a new proposer is randomly selected. The data block may be implemented in a data structure as shown in FIG. 3. In some embodiments of the present application, these data blocks may be linked to form the current public transaction record linked list in a linked list manner as shown in the drawings, which means that head of the linked list
15 (data header) of each data block includes a digital signature of a previous data block (a data block 1 including a linked list header or head of the linked list), so that each data block of the current public transaction record linked list may be verified conveniently, thus being capable of quickly determine whether the current public transaction record linked list has been maliciously damaged, and being capable of switching back to a normal state in time by deleting any illegal data block. Other suitable
20 data structures may be readily selected by those skilled in the art, not to be limited by the present application, to record each data block and the current public transaction record linked list, thus being capable of conveniently verifying whether the current public transaction record linked list is maliciously damaged, and being capable of switching back to the normal state in time in the case that the current public transaction record linked list is maliciously damaged, which is not further limited
25 by the present disclosure. Similarly, the present invention does not limit the verification means. The verification means may be a conventional verification method in the field (for example, verifying the digital signature in the data block is correct) and a customized standard (for example, verifying whether an amount of each transaction exceeds an upper limit, and whether the historical transaction record and the transaction method are legal). Those skilled in the art may select an appropriate
30 verification method according to an actual situation. In one or more embodiments of the present application, the current public transaction record linked list may be recorded in a data center, and each data block passing the verification added to the linked list being stored in the data center may be retrieved by any node (including a verification node and a non-verifying ordinary node) at any time after being added to the current public transaction record linked list stored in the data center.

Alternatively, in other embodiments of the present application, each node may also store locally a copy of the current public transaction record linked list, and after a data block passes the verification, the data block is updated to the local copy of the current public transaction record linked list, thus a decentralized storage of the current public transaction record linked list is implemented and improving a robustness and a security of the system. According to one or more embodiments of the present invention, in order to ensure that only one data block is added to the record linked list at a time, the data block which has been verified by the majority of the verification nodes, but the data block is yet to be added to the record linked list, selection of a new data block is stopped.

Since the IBFT consensus aims to establish a trust between nodes in a distrustful network environment, the above network transaction verification method using the IBFT consensus allows current users of some verification nodes to be malicious, who may collude to make wrong decision on transaction verification and disrupting a normal transaction to maximum extent. Even in a worst case scenario, the IBFT consensus is still capable to ensure that every transactions recorded in the current public transaction record linked list are verified by most of "loyal" verification nodes in the network and becomes a part of permanent records, and may track a verification result made by each verification node at any time, or further serves as a basis for dynamically adjusting the verification nodes in subsequent transaction verification processes. Therefore, on one hand, a verification method based on proof of authority can verify the security of a transaction faster than a verification method based on a proof of work used by a traditional blockchain technology to meet actual needs; and on the other hand, the verification method of the proof of authority retains an advantage that the verification method of the proof of work may be applied to an untrusted network environment.

In order to further deal with an explosively increased in transaction volume in a specific period, with reference to a sub-flow chart shown in FIG. 4, in one or more embodiments of the present application, when a number of the verification nodes is $3F+1$, the above step of initializing the public P2P channel transmission data between the plurality of nodes and the current public transaction record linked list for recording the network transaction further includes the following sub-steps of: collecting verification results returned by at least $2F+1$ number of verification nodes; and counting the collected verification results, and when a data blocks passes verification by greater than or equal to $F+1$ number of verification nodes, recording the data block to an end of the current public transaction record linked list. F is a number of allowable error/malicious nodes in a network.

In practice, since the IBFT consensus assumes that the "loyal" verification nodes in a network would all make the same verification results at any time, and a number of the malicious verification nodes of the current users does not exceed $1/3$ of a total number of the nodes, when receiving verification results returned by more than $2F+1$ number of verification nodes, it is considered that

even in the worst scenario (which means that verification results made by all malicious verification nodes are received), sufficient correct verification results from "loyal" verification nodes can be received for a transaction, thus false decision on the security of a transaction can be avoided. Further, in order to ensure that every "loyal" verification node would make the same verification result, without having a potential hidden danger that a "dishonest" intermediate node modifies transmitted information, as described in a traditional Byzantine Generals Problem, the data block is write-locked during the process of being verified, so that all the nodes process a data block in a read-only manner. At the same time, when a data block is being transmitted, even if the data block passes by a node that is currently occupied by a malicious user, the data block cannot be tampered by the malicious user.

In one or more embodiments of the present application, in order to ensure a security of the system, data blocks are being generated at a constant rate. When there is no new transaction information, transaction information in a corresponding data block is empty. In fact, since the length of the current public transaction record linked list is increased steadily, and a private key is used as the signature by each verification nodes may be updated periodically according to the length of the current public transaction record linked list, any malicious user who attempts to collect the signatures in the data blocks needs to update the private key simultaneously with the verification nodes, so that the security of the system is enhanced.

In order to identify and eliminate the malicious verification nodes of the current users which are assumed by the IBFT consensus which account for 1/3 of the total number of the nodes, in one or more embodiments of the present application, when the number of the data blocks added to the current public transaction record linked list is greater than an updating threshold, the following step is executed: verifying the verification nodes, and when a verification node fails to pass verification by 2/3 of a total number of all the verification nodes, deselecting the verification node from being used as one of the verification nodes. Accordingly, in order to replace the deselected verification node, in one or more embodiments of the present application, when the number of the data blocks added to the current public transaction record linked list is greater than the updating threshold, the following step is executed: when a newly added verification node passes verification by 2/3 of a total number of all the verification nodes, selecting the newly added node as one of the verification nodes. Specifically, in some embodiments of the present invention, when 30,000 data blocks (including data blocks with empty transaction information) are added to the current public transaction record linked list, the above update of notification nodes take place. In one or more embodiments of the present application, the above verification of a verification node may also be based on the accuracy of said verification node in verifying the data block in the current public transaction record linked list. Specifically, the last 1/3 verification nodes in order of verification accuracy would be eliminated. In

one or more embodiments of the present invention, when a node to be verified has not been verified by certain node, maintaining the node as a node to be verified.

Since the data block on the current public transaction record linked list can be viewed by all the nodes, a private transaction transmission channel is established between some mutually trusted transaction parties to verify the security of the transaction between them. Specifically, with reference to a sub-flow chart shown in FIG. 5, the following steps are being executed between any two or more nodes with high mutual trust: initializing a private transaction transmission channel (such as specifying a port of the private transaction transmission channel between any two or more nodes) and a private transaction data linked list based on the current public transaction record linked list (such as creating a head of the private transaction data linked list based on the current public transaction record linked list), and sharing a private transaction data link to the two or more nodes through the private transaction transmission channel; by means of a Raft consensus, adding data block containing the transaction information and the signature between the two or more nodes to end of the private transaction data linked list; and through the private transaction transmission channel, updating the private transaction data linked list to the two or more nodes; wherein the private transaction transmission channel is separated from the public P2P channel. Data can only be transmitted between any two or more nodes through the private transaction transmission channel. Further, the current public transaction record linked list and the private transaction transmission channel are separated from each other. There is no interoperability or intersection between the two. Therefore, benefits of transmitting transaction through the private transaction transmission channel at least include being able not to broadcast transaction information to the whole network. Each private transaction transmission channel may be deemed as a discrete Raft network.

Running of the private transaction transmission channel indicates that although network participants may not trust each other completely (the network participants may even be competitors in the same industry), the network participants may select other participants to conduct a private transaction with, and only transmit the data block on the private transaction transmission channel, so that only parties using the private transaction transmission channel can see transaction data between the parties, which provides an option for higher transaction privacy. It can be seen from previous discussion that since all parties who conduct the transactions through the private transaction transmission channel have high trust in one another, these nodes would use transaction verification means in a different way from the current public transaction record linked list, which means that the Raft consensus is used instead of the IBFT consensus. Concurrently, each node may have an election timeout for a current Raft network. In one embodiment of the present application, time period of the election timeout is randomly between 150 milliseconds and 300 milliseconds. In other embodiments

of the present application, the time period is 160 milliseconds, 180 milliseconds, 200 milliseconds, 220 milliseconds, 250 milliseconds, or 280 milliseconds. When running the Raft network, each node executes at least the following two activities: leader election and log replication. At the beginning of the leader election, all the nodes in the Raft network are initialized in a “follower” state. After timeout
5 period of a certain node is over, a state of the node changes to a “candidate” state, and voting is proposed. The candidate node transmits a “Request Vote” to several other nodes. If at least a half of other nodes agree with voting, the state of the node is changed from the candidate state to a “leader” state, and after a short period of time, a heartbeat message is transmitted to all the nodes in the candidate state to maintain the current state of all the nodes. The node in the candidate state resets its
10 own election timeout period after receiving a heartbeat message of the node in the leader state. When a client transmits a request to a node in the leader state, relevant data is written in a local log by the node in the leader state. Then, the node in the leader state transmits a request for update of said local log to other nodes in the follower state. When the data are written to the local log by other nodes in the follower state, and a success message is returned to the node in the leader state. If the node in the
15 leader state receives the success message returned by more than a half of the nodes, relevant data will be set to be in a confirmed state in the local log of the node in the leader state. Then, the node in the leader state transmits a confirmation message to other nodes in the follower state, so that setting of corresponding data of the nodes in the follower state in the local log is changed into the confirmed state, thus completing a complete log replication process.

20 Meanwhile, a transaction processing speed can increase to thousands of transactions per second to meet daily transaction needs. For example, in the embodiment of the present invention, at least 1,400 transactions are processed per second. In other embodiments, at least 1,650 transactions are processed per second. In addition, robustness of the Raft consensus enables transaction between parties can be verified as usual as long as more than half of the two or more nodes participating in the
25 transaction are running normally. Moreover, the private transaction data linked list can be consistent on each node through operations such as rollback after the corresponding node is restored. In other words, the Raft consensus is based on a trust network, corresponding Raft network tolerate less than 1/2 of the nodes as error/malicious nodes. In one or more embodiments of the present invention, when most of the nodes fail, the private transaction will be stopped, and the corresponding data block of
30 the private transaction will not be added to the private transaction data block linked list. For example, when two nodes conduct a private transaction, and one of the nodes fails or is off-line, the transaction will be stopped. In one or more embodiments of the present invention, an off-line node may rejoin an original private transaction later, and the missing data linked list when being off-line is updated to the rejoined node.

Based on high throughput nature, security and privacy brought by the present invention, transaction between multiple business organizations can be effectively managed or processed. The technical solutions of the present invention may be used to manage or process transactions between multiple commercial organizations, between multiple commercial organizations and consumers, and/or between multiple consumers. In one embodiment of the present invention, the above technical solutions of the present invention may be applied to managing a consumption reward plan. The business organizations and the consumers are the plurality nodes of the network transaction. Consumption records, specific bonus points set by the business organizations, universal points, and the like are the transaction information and the signature in the data block.

With reference to a frame diagram of a network transaction verification system based on a plurality of nodes shown in FIG. 6, those of ordinary skills in the art would appreciate that the units and algorithm steps described in each embodiments of the present application may be combined and can be implemented by electronic hardware, or a combination of computer software and electronic hardware. Whether these functions are executed by hardware or software depends on the specific application and design constraint of the technical solution of the present invention for each specific application. Those skilled in the art can use different methods to implement the described functions for each specific application, but such implementation is not being out of the scope of the present invention.

In the embodiments provided by the present invention, it should be understood that the disclosed apparatus and method may be implemented in other ways. For example, the system embodiments described above are mere illustrative. For example, division of the modules or units is only one logical function division. In practice, there may be other methods of division. For another example, multiple units or components may be combined or integrated into another system, or some features may be omitted or not executed. In addition, the illustrated or discussed mutual coupling or direct coupling or communication connection may be indirect coupling or communication connection through some interfaces, apparatuses or units, and may be in electrical, mechanical or other forms.

The units illustrated as separation parts may either be separated or not physically separated, and the parts displayed as units may either be physical units or not physical units, i.e., the parts displayed as units may be located in one place or distributed to multiple network units. Some or all of the units may be selected according to actual needs to achieve the objectives of the solutions in the embodiments. Some or all of the modules may be selected according to actual needs to achieve the objectives of the solutions in the embodiments.

In addition, each functional unit in each embodiment of the present invention may be integrated into one processing unit, or each unit may exist alone physically, or two or more units may be

integrated into one unit. The integrated units above may be implemented in the form of hardware, or in the form of software functional units.

The integrated units, if being implemented in the form of a software functional unit and sold or used as an independent product, may also be stored in a computer-readable storage medium. Based on such understanding, all or a part of the steps of the methods in the above embodiments of the present invention may also be implemented by instructing relevant hardware through a computer program. The program may be stored in a computer-readable storage medium. The computer program, when executed by the processor, may implement the steps of each method embodiment mentioned above. The computer program includes a computer program code, which may be in the form of a source code, an object code, an executable file or some intermediate form. The computer-readable medium may include any entity or apparatus, recording medium, USB disk, mobile hard disk, magnetic disk, optical disk, computer memory, ROM (Read-Only Memory), RAM (Random Access Memory), electrical carrier signal, telecommunication signal and software distribution medium capable of carrying the computer program code. It should be noted that the contents contained in the computer-readable medium may be appropriately increased or decreased according to the requirements of legislation and patent practice in jurisdictions. For example, in some jurisdictions, according to legislation and patent practice, the computer-readable medium does not include electrical carrier signals and telecommunication signals.

The above embodiments are only used to illustrate the technical solution of the present invention, rather than limiting the present invention; although the present invention has been described in detail with reference to the foregoing embodiments, those of ordinary skill in the art should understand that: the technical solutions described in the foregoing embodiments can be still modified or some of the technical features can be equivalently replaced; however, these modifications, changes or replacements do not make the essence of the corresponding technical solutions deviate from the spirit and scope of the technical solutions of each embodiment of the present invention, and should all be covered within the protection scope of the present invention.

CLAIMS

1. A network transaction verification method based on a plurality of nodes, comprising the following steps of:

5 identifying authenticated identities of a plurality of nodes participating in the network transaction;

selecting a set of nodes from the plurality of nodes as verification nodes for verifying nodes in the network transaction;

10 initializing a public P2P channel transmission data between the plurality of nodes and a current public transaction record linked list for recording the network transaction;

generating and adding, by means of an IBFT consensus, a data block containing transaction information and signature to the current public transaction record linked list; and

15 after a number of data blocks added to the current public transaction record linked list is greater than an updating threshold, updating the verification nodes, wherein the updating step comprises verification of verification nodes.

2. The network transaction verification method according to claim 1, wherein when a number of the verification nodes is $3F+1$, the step of initializing the public P2P channel transmission data between the plurality of nodes and the current public transaction record linked list for recording the network transaction further comprises the following sub-steps of:

20 collecting verification results returned by at least $2F+1$ number of verification nodes; and

counting the collected verification results, and when a data block passes verification by greater than or equal to $F+1$ number of verification nodes, recording the data block to end of the current public transaction record linked list.

25 3. The network transaction verification method according to claim 1, wherein the data block is generated at a constant rate, and when no new transaction information occurs, transaction information in corresponding data block is empty.

4. The network transaction verification method according to claim 1, wherein the updating step further comprises:

30 when a verification node fails to pass verification by $2/3$ of a total number of all the verification nodes, deselecting said verification node from being one of the set of verification nodes.

5. The network transaction verification method according to claim 1, wherein the updating step further comprises:

when a newly added node passes verification by 2/3 of a total number of all the verification nodes, selecting the newly added node as one of the set of verification nodes.

6. The network transaction verification method according to claim 1, wherein the updating threshold is 30,000.

7. The network transaction verification method according to claim 1, wherein the following steps are being executed between any two or more nodes:

initializing a private transaction transmission channel and a private transaction data linked list based on the current public transaction record linked list, and sharing a private transaction data link to the two or more nodes through the private transaction transmission channel;

by means of a Raft consensus, adding data block containing the transaction information and the signature between the two or more nodes to end of the private transaction data linked list; and

through the private transaction transmission channel, updating the private transaction data linked list to the two or more nodes;

wherein, the private transaction transmission channel is separated from the public P2P channel.

8. The network transaction verification method according to claim 1, wherein the data block is write-locked in the process of being verified, so that the data block is being processed by all the nodes in a read-only manner.

9. A computer system, comprising a memory, a processor, and a computer program stored on the memory and executable on the processor, wherein the processor implements the method according to any one of claims 1 to 8 when executing the computer program.

10. A storage medium, wherein the storage medium stores a computer program, and the computer program is configured to execute the method according to any one of claims 1 to 8 during running.