

DATA PROCESSING METHOD AND APPARATUS

TECHNICAL FIELD

[01] The present invention relates to the field of computer technologies, and in particular, to a data processing method and apparatus.

5 BACKGROUND

[02] A "blockchain" technology is a special distributed database technology originally designed by a person using the name "Satoshi Nakamoto" for a digital currency, Bitcoin. The blockchain is suitable to store simple and sequential data that can be verified within the system, and ensures, based on cryptography and a consensus algorithm, that the stored data
10 cannot be tampered with or forged and is public and transparent across the network.

[03] Because an asset and a transaction record in each account are public, when an account address of an account is obtained by using any other account, all assets and historical transactions in the account can be obtained by querying the blockchain.

[04] It is necessary to provide a more reliable solution.

15 SUMMARY

[05] Embodiments of the present specification provide a data processing method and apparatus, to resolve a problem that an existing blockchain technology is prone to a data leakage.

[06] An embodiment of the present specification provides a data processing method,
20 including:

[07] determining, by a transaction initiator, information to be hidden in transaction data;

[08] obtaining, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function;

[09] constructing, according to the transaction root, proof data of validity of the information
25 to be hidden; and initiating a transaction request to write the proof data and the transaction root carrying a signature of the transaction initiator on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

[10] Optionally, before using the transaction data as the input of the predetermined one-way

function, the method further includes:

- [11] determining pre-transaction data and post-transaction data in the transaction data, and
- [12] obtaining, by using the transaction data as the input of the predetermined one-way function, the transaction root that is output by the predetermined one-way function includes:
- 5 [13] obtaining, by using the pre-transaction data as an input of a first predetermined one-way function, an input transaction root that is output by the first predetermined one-way function;
- [14] obtaining, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function; and
- 10 [15] using the obtained input transaction root and output transaction root as to-be-chained transaction roots.
- [16] Optionally, obtaining, by using the post-transaction data as the input of a second predetermined one-way function, the output transaction root that is output by the second predetermined one-way function includes:
- 15 [17] constructing, according to the post-transaction data and a random number, a bottom layer of a predetermined one-way function tree, and determining an output transaction root corresponding to a top layer of the predetermined one-way function tree.
- [18] Optionally, constructing, according to the transaction root, proof data of validity of the information to be hidden includes:
- 20 [19] performing, by using the transaction root and the information to be hidden as inputs, a predetermined logical operation; and
- [20] Performing, by using the information to be hidden as a protected object, a zero-knowledge proof on the logical operation to obtain the proof data.
- [21] An embodiment of the present specification provides a data processing method,
- 25 including:
- [22] receiving a transaction request from a transaction initiator, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;
- 30 [23] performing verification on the transaction root, to obtain a first verification result; performing, according to the transaction root, verification on the proof data to obtain a second verification result; and
- [24] approving or rejecting the transaction request according to the first verification result and the second verification result.

[25] Optionally, performing verification on the transaction root includes:

[26] verifying whether the transaction root has a valid signature.

[27] Optionally, verifying whether the transaction root has a valid signature includes:

[28] determining an address associated with an output transaction root in the transaction root,

5 where the output transaction root is constructed according to post-transaction data in the transaction data; and

[29] verifying, if determining that a quantity of obtained addresses is 1, whether the transaction root has a signature corresponding to the transaction initiator.

[30] Optionally, performing verification on the transaction root includes:

10 [31] verifying whether an output transaction root in the transaction root is the same as an already-chained output transaction root and input transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data.

[32] Optionally, performing verification on the transaction root includes:

[33] verifying whether an input transaction root in the transaction root is the same as an

15 already-chained output transaction root and input transaction root, where the input transaction root is constructed according to pre-transaction data in the transaction data.

[34] Optionally, approving or rejecting the transaction request according to the first verification result and the second verification result includes:

[35] if the first verification result and the second verification result both are successful,

20 writing, in response to the transaction request, the transaction root carrying a signature of the transaction initiator and the proof data on the blockchain; or

[36] if the first verification result and/or the second verification result is failed, rejecting the transaction request.

[37] An embodiment of the present specification provides a data processing method,

25 including:

[38] determining, by a transaction initiator, information to be hidden in transaction data;

[39] obtaining, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function;

[40] constructing, according to the transaction root, proof data of validity of the information

30 to be hidden; and sending the transaction root and the proof data to a transaction receiver, wherein the transaction receiver performs verification on the transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain.

[41] An embodiment of the present specification provides a data processing method, including:

[42] receiving, by a transaction receiver, a transaction root and proof data from a transaction sender, where the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[43] performing verification on the transaction root according to the information to be hidden and the predetermined one-way function; and

[44] if a verification result is successful, initiating a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

[45] An embodiment of the present specification provides a data processing method, including:

[46] receiving a transaction request from a transaction receiver, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data initiated by a transaction initiator as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[47] performing verification on the transaction root, to obtain a first verification result; performing, according to the transaction root, verification on the proof data to obtain a second verification result; and

[48] approving or rejecting the transaction request according to the first verification result and the second verification result.

[49] Optionally, performing verification on the transaction root includes:

[50] determining an address associated with an output transaction root in the transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data; and

[51] if determining that a quantity of obtained addresses is greater than or equal to a predetermined threshold, verifying whether the transaction root carries signatures corresponding to the transaction initiator and the transaction receiver.

[52] An embodiment of the present specification provides a data processing apparatus, including:

[53] a determining unit, configured to determine information to be hidden in transaction data;

[54] a processing unit, configured to obtain, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function, and construct, according to the transaction root, proof data of validity of the information to be hidden; and

[55] an initiation unit, configured to initiate a transaction request to write the proof data and the transaction root carrying a signature of the transaction initiator on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

[56] Optionally, the determining unit is further configured to determine pre-transaction data and post-transaction data in the transaction data, and

[57] the processing unit is specifically configured to obtain, by using the pre-transaction data as an input of a first predetermined one-way function, an input transaction root that is output by the first predetermined one-way function; obtain, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function; and use the obtained input transaction root and output transaction root as to-be-chained transaction roots.

[58] Optionally, the processing unit is specifically configured to construct, according to the post-transaction data and a random number, a bottom layer of a predetermined one-way function tree, and determine an output transaction root corresponding to a top layer of the predetermined one-way function tree.

[59] Optionally, the processing unit is specifically configured to perform a predetermined logical operation by using the transaction root and the information to be hidden as inputs; and perform, by using the information to be hidden as a protected object, a zero-knowledge proof on the logical operation to obtain the proof data.

[60] An embodiment of the present specification provides a data processing apparatus, including:

[61] a receiving unit, configured to receive a transaction request from a transaction initiator, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[62] a verification unit, configured to perform verification on the transaction root to obtain a first verification result, and perform, according to the transaction root, verification on the

proof data to obtain a second verification result; and

[63] a response unit, configured to approve or reject the transaction request according to the first verification result and the second verification result.

[64] Optionally, the verification unit is specifically configured to verify whether the transaction root has a valid signature.

[65] Optionally, the verification unit is specifically configured to determine an address associated with an output transaction root in the transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data; and if determining that a quantity of obtained addresses is 1, verify whether the transaction root has a signature corresponding to the transaction initiator.

[66] Optionally, the verification unit is specifically configured to verify whether an output transaction root in the transaction root is the same as an already-chained output transaction root and input transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data.

[67] Optionally, the verification unit is specifically configured to verify whether an input transaction root in the transaction root is the same as an already-chained output transaction root and input transaction root, where the input transaction root is constructed according to pre-transaction data in the transaction data.

[68] Optionally, the response unit is specifically configured to: if the first verification result and the second verification result both are successful, write, in response to the transaction request, the transaction root carrying a signature of the transaction initiator and the proof data on the blockchain; or if the first verification result or the second verification result is failed, reject the transaction request.

[69] An embodiment of the present specification provides a data processing apparatus, including:

[70] a determining unit, configured to determine, by a transaction initiator, information to be hidden in transaction data;

[71] a processing unit, configured to obtain, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function, and construct, according to the transaction root, proof data of validity of the information to be hidden; and

[72] a sending unit, configured to send the transaction root and the proof data to a transaction receiver, wherein the transaction receiver performs verification on the transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data and the

transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain.

[73] An embodiment of the present specification provides a data processing apparatus, including:

5 [74] a receiving unit, configured to receive a transaction root and proof data from a transaction sender, where the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[75] a verification unit, configured to perform verification on the transaction root according to the information to be hidden and the predetermined one-way function; and

10 [76] an initiation unit, configured to: if a verification result is successful, initiate a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

15 [77] An embodiment of the present specification provides a data processing apparatus, including:

[78] a receiving unit, configured to receive a transaction request from a transaction receiver, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data initiated by a transaction initiator as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

20 [79] a verification unit, configured to perform verification on the transaction root, to obtain a first verification result, and perform, according to the transaction root, verification on the proof data to obtain a second verification result; and

[80] a response unit, configured to approve or reject the transaction request according to the first verification result and the second verification result.

25 [81] Optionally, the verification unit is specifically configured to determine an address associated with an output transaction root in the transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data; and if determining that a quantity of obtained addresses is greater than or equal to a predetermined threshold, verify whether the transaction root carries signatures corresponding to the transaction initiator and the transaction receiver.

30 [82] An embodiment of the present specification provides a data processing apparatus,

including:

[83] a processor; and

[84] a memory configured to store a computer executable instruction, where when executed, the executable instruction enables the processor to perform the following operations:

5 [85] determining information to be hidden in transaction data;

[86] obtaining, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function;

[87] constructing, according to the transaction root, proof data of validity of the information to be hidden; and initiating a transaction request to write the proof data and the transaction
10 root carrying a signature of the transaction initiator on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

[88] An embodiment of the present specification provides a data processing apparatus, including:

15 [89] a processor; and

[90] a memory configured to store a computer executable instruction, where when executed, the executable instruction enables the processor to perform the following operations:

[91] receiving a transaction request from a transaction initiator, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction
20 root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[92] performing verification on the transaction root, to obtain a first verification result; performing, according to the transaction root, verification on the proof data, to obtain a second verification result; and

25 [93] approving or rejecting the transaction request according to the first verification result and the second verification result.

[94] An embodiment of the present specification provides a data processing apparatus, including:

[95] a processor; and

30 [96] a memory configured to store a computer executable instruction, where when executed, the executable instruction enables the processor to perform the following operations:

[97] determining information to be hidden in transaction data;

[98] obtaining, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function;

[99] constructing, according to the transaction root, proof data of validity of the information to be hidden; and sending the transaction root and the proof data to a transaction receiver, wherein the transaction receiver performs verification on the transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain.

[100] An embodiment of the present specification provides a data processing apparatus, including:

[101] a processor; and
[102] a memory configured to store a computer executable instruction, where when executed, the executable instruction enables the processor to perform the following operations:

[103] receiving a transaction root and proof data from a transaction sender, where the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[104] performing verification on the transaction root according to the information to be hidden and the predetermined one-way function; and

[105] if a verification result is successful, initiating a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

[106] An embodiment of the present specification provides a data processing apparatus, including:

[107] a processor; and
[108] a memory configured to store a computer executable instruction, where when executed, the executable instruction enables the processor to perform the following operations:

[109] receiving a transaction request from a transaction receiver, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data initiated by a transaction initiator as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[110] performing, according to the transaction root, verification on the transaction root, to

obtain a first verification result; performing verification on the proof data to obtain a second verification result; and

[111] approving or rejecting the transaction request according to the first verification result and the second verification result.

5 [112] The foregoing at least one technical solution used in the embodiments of the present specification can achieve the following beneficial effects:

[113] As can be seen, in the embodiments, at first, the one-way function is used to ensure that the transaction data is authentic and cannot be tampered with. Then the proof data is constructed to prove validity of the information to be hidden without revealing any
10 information to be hidden to a non-transaction participant. Compared with a solution in existing technologies in which a data leakage of any node causes a leakage of all data, in the embodiments, only data of the node is leaked, and an attacker still cannot view private data of other nodes. Therefore, data security of the blockchain can be improved.

BRIEF DESCRIPTION OF THE DRAWINGS

15 [114] Accompanying drawings described herein are used to provide further understanding about the present invention, and constitute one portion of present invention. Exemplary embodiments of the present invention and their descriptions are used to explain the present invention, and do not constitute an inappropriate limitation on the present invention. In the figures:

20 [115] FIG. 1a and FIG. 1b are schematic diagrams of application scenarios of a data processing method, according to an embodiment of the present specification;

[116] FIG. 2 is a schematic flowchart of a data processing method, according to Embodiment 1 of the present specification;

[117] FIG. 3 is a schematic flowchart of a data processing method, according to
25 Embodiment 2 of the present specification;

[118] FIG. 4 is a schematic flowchart of a data processing method, according to Embodiment 3 of the present specification;

[119] FIG. 5 is a schematic flowchart of a data processing method, according to Embodiment 4 of the present specification;

30 [120] FIG. 6a is a schematic flowchart of constructing a transaction root in a data processing method, according to Embodiment 5 of the present specification;

[121] FIG. 6b is a schematic flowchart of constructing proof data in a data processing

method, according to Embodiment 5 of the present specification;

[122] FIG. 7 is a schematic structural diagram of a data processing apparatus, according to Embodiment 6 of the present specification;

[123] FIG. 8 is a schematic structural diagram of a data processing apparatus, according to Embodiment 7 of the present specification;

[124] FIG. 9 is a schematic structural diagram of a data processing apparatus, according to Embodiment 8 of the present specification;

[125] FIG. 10 is a schematic structural diagram of a data processing apparatus, according to Embodiment 9 of the present specification;

[126] FIG. 11 is a schematic structural diagram of a data processing apparatus, according to Embodiment 10 of the present specification; and

[127] FIG. 12 is a schematic structural diagram of an electronic device, according to Embodiment 11 of the present specification.

DETAILED DESCRIPTION OF THE INVENTION

[128] To make the objectives, technical solutions, and advantages of the present invention clearer, the technical solutions of the present invention will be described below with reference to embodiments of the present invention and corresponding accompanying drawings. Apparently, the described embodiments are merely a part rather than all of the embodiments of the present invention. All other embodiments obtained by a person of ordinary skill in the art based on the embodiments of the present invention without creative effects shall fall within the protection scope of the present invention.

[129] It should be noted that, according to a one-way function, it is easy to calculate $f(x)=y$ for any x in the domain of f , but for any y in the codomain of f , it is impossible to calculate x so that $y=f(x)$. That is, it is easy to calculate an output for any input, but it is difficult to deduce the input from the output.

[130] In a zero-knowledge proof, a prover enables a verifier to believe that some inference is correct without providing any useful information (protected/hidden information) to the verifier.

[131] Referring to FIG. 1a, an application scenario of the present invention may be as follows:

[132] A node A on a blockchain needs to transfer some money from a first account of a transaction initiator corresponding to the node A to a second account of the transaction

initiator, or merge the first account and the second account. Before requesting to write the transfer/merging in a form of a transaction on the blockchain, the node A first determines information to be hidden, for example, inputs (such as pre-transaction balances) and outputs (such as post-transaction balances) of the first account and the second account, in transaction data of the transaction. Then, the node A calculates a transaction root of the transaction, and constructs, based on the transaction root, proof data of the information to be hidden, to prove validity, authenticity, and other information to be hidden. Finally, the node A requests to write the transaction root and the proof data in a form of a transaction on the blockchain.

[133] Understandably, the blockchain allows one person to apply for a plurality of accounts, but according to a signature mechanism of the blockchain, each person is allocated a unique corresponding signature. Therefore, when a transaction occurs between different accounts of a person A, the transaction requires only a signature of A.

[134] Referring to FIG. 1b, another application scenario of the present invention may be as follows:

[135] A node A on a blockchain needs to transfer some money from a transaction initiator to a transaction receiver corresponding to a node B. Before requesting to write the transfer in a form of a transaction on the blockchain, the node A first determines information to be hidden, for example, inputs and outputs of the transaction initiator and the transaction receiver and balances of the transaction initiator and the transaction receiver, in transaction data of the transaction. Then, the node A calculates a transaction root of the transaction, and constructs, based on the transaction root, proof data of the information to be hidden, to prove validity, authenticity, and other information to be hidden. Finally, the node A sends the transaction root and the proof data to the transaction receiver, so that the transaction receiver performs verification on the transaction root. If the verification succeeds, after signing the transaction root, the transaction receiver initiates a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on the blockchain. Alternatively, after signing the transaction root, the transaction receiver returns the signed transaction root plus a verification result to the transaction initiator, to instruct the transaction initiator to initiate a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on the blockchain.

[136] In the above-mentioned two scenarios, the transfer or merging is an example for ease of understanding the "transaction." In fact, the "transaction" may alternatively be recording a piece of information in the node A, sending a piece of information to the node B

from the node A, etc. Correspondingly, the "information to be hidden" may be set freely. For example, the "information to be hidden" may be, but not limited to, balances of the transaction between the two parties.

[137] Technical solutions provided in embodiments of the present invention are described
5 below in detail with reference to the accompanying drawings.

[138] Embodiment 1

[139] FIG. 2 is a schematic flowchart of a data processing method, according to Embodiment 1 of the present specification. The method may be performed by a participated transaction node in a blockchain, and may specifically include the following steps:

10 [140] Step 22: A transaction initiator determines information to be hidden in transaction data.

[141] It should be noted that, an implementation of step 22 may be as follows:

[142] A node obtains a transaction inventory completed by a transaction initiator (user). The transaction inventory records an item and content corresponding to the item, for example,
15 a payee, a payer, pre-transaction and post-transaction balances of the payee, and pre-transaction and post-transaction balances of the payer. Next, the node selects corresponding information from the transaction inventory based on a predetermined hiding rule.

[143] The hiding rule may be: determining, based on a preset hidden item, an item in the
20 transaction inventory corresponding to the hidden item and content of the item as information to be hidden. For example, the hidden item includes the pre-transaction and post-transaction balances of the payee and the pre-transaction and post-transaction balances of the payer.

[144] Alternatively, the hiding rule may be: selecting data corresponding to a specific key word from the transaction inventory based on a preset screening rule. For example, if the
25 specific key word is "balance" or "transaction amount," a corresponding number is selected, and is used as information to be hidden.

[145] It should be noted that, another implementation of step 22 may be as follows:

[146] When completing a transaction inventory, a transaction initiator customizes and labels data to be hidden in the transaction inventory. For example, pre-transaction and
30 post-transaction balances of a payee and pre-transaction and post-transaction balances of a payer are labeled.

[147] After obtaining the transaction inventory completed by the transaction initiator, a node determines information to be hidden in transaction data based on the label.

[148] Step 24: Obtain, by using the transaction data as an input of a predetermined

one-way function, a transaction root that is output by the predetermined one-way function, and construct, according to the transaction root, proof data of validity of the information to be hidden.

[149] It should be noted that, an implementation of constructing a transaction root in step 5 24 may be as follows:

[150] First, the node analyzes the transaction data based on the transaction inventory corresponding to the transaction data, to determine pre-transaction data and post-transaction data in the transaction data. For example, the pre-transaction and post-transaction balances of the payee and the pre-transaction and post-transaction balances of the payer are determined.

10 [151] Next, the node obtains, by using the pre-transaction data as an input of a first predetermined one-way function, an input transaction root that is output by the first predetermined one-way function; and obtains, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function. The first predetermined one-way function and the 15 second predetermined one-way function may be the same one-way function.

[152] The one-way function may be a one-way function tree. Correspondingly, the step of constructing an input transaction root/output transaction root may be specifically: constructing a bottom layer of a predetermined one-way function tree according to the pre-transaction and post-transaction data and a random number, and determining an input 20 transaction root/output transaction root corresponding to a top layer of the predetermined one-way function tree. For example, a random number of a fixed length is added behind the balance of the pre-transaction and post-transaction. For example, when a 128-bit number is defined, the first 10 bits may represent the balance, and bits behind may represent a random number.

25 [153] Understandably, there are at least one input transaction root and at least one output transaction root. For example, in root A (a leaf node A has an account balance 100), root B (a leaf node B has an account balance 800), root A' (the leaf node A has an account balance 50), root B' (the leaf node B has an account balance 950), root A and root B are input transaction roots, and root A' and root B' are output transaction roots. In addition, the random number is 30 added during the transaction root construction to distinguish each input transaction root and output transaction root, to prevent a collision subsequently when a consensus node performs verification on the input transaction root and output transaction root.

[154] An implementation of constructing proof data in step 24 may be as follows:

[155] A predetermined logical operation is performed by using the transaction root and

the information to be hidden as inputs; and a zero-knowledge proof is performed on the logical operation by using the information to be hidden as a protected object, to obtain the proof data.

[156] It should be noted that, the predetermined logical operation may be a particular function customized according to content to be proved. For example, when a transaction between a first account and a second account of the transaction initiator requires to be proved, the particular function may specifically perform calculation by using a transaction root and information to be hidden as inputs, to determine whether pre-transaction balances of the first account and the second account are the same as post-transaction balances of the first account and the second account, a difference between a pre-transaction balance and a post-transaction balance does not exceed n dollars, or interest of the second account after the transaction is increased, etc. Next, a proof is output for an entire function operation process in a zero-knowledge proof manner, to prove that the hidden information can be used to calculate a particular function.

[157] Step 26: Initiate a transaction request to write the proof data and the transaction root carrying a signature of the transaction initiator on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

[158] Accordingly, in this embodiment, at first, the one-way function is used to ensure that the transaction data is authentic and cannot be tampered with. Next, the proof data is constructed to prove validity of the information to be hidden without revealing any information to be hidden to a non-transaction participant. Compared with existing technologies, this embodiment can effectively prevent data privacy of the entire blockchain from being affected by a data leakage of any node, and improve data security of the blockchain.

[159] Embodiment 2

[160] FIG. 3 is a schematic flowchart of a data processing method, according to Embodiment 2 of the present specification. The method may be performed by a consensus node in a blockchain, and may specifically include the following steps:

[161] Step 32: Receive a transaction request from a transaction initiator, where the transaction request is used to request to write a transaction root and proof data on a blockchain. The transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data.

[162] It should be noted that, step 32 may be a step performed based on Embodiment 1. Details of the transaction request, the transaction root, and the proof data in this step can be referred to relevant descriptions in Embodiment 1, and thus, are not described herein again.

[163] Step 34: Perform verification on the transaction root to obtain a first verification
5 result, and perform, according to the transaction root, verification on the proof data to obtain a second verification result.

[164] An implementation of performing verification on the transaction root in step 34 may be as follows:

[165] verifying whether the transaction root has a valid signature. If the transaction root
10 has a valid signature, the first verification result is successful. If the transaction root does not have a valid signature, the first verification result is failed.

[166] Specifically, an address associated with an output transaction root in the transaction root is first determined. The output transaction root is constructed according to post-transaction data in the transaction data. If it is determined that a quantity of obtained
15 addresses is 1, whether the transaction root has a signature corresponding to the transaction initiator will be verified.

[167] Understandably, each output transaction root corresponds to an address hash(pk), and a quantity of transaction participants may be determined based on the addresses; next, whether the transaction root has valid signatures of all participants will be verified. For
20 example, when there is only one address, the transaction may be considered as a transaction between different accounts of the same user. Furthermore, if a signature of the user is verified to be correct, the first verification result is determined to be successful.

[168] In addition, the signature verification is related to a digital signature mechanism of the blockchain, which is not limited herein.

[169] Another implementation of performing verification on the transaction root in step
25 34 may be as follows:

[170] Based on the foregoing implementation of performing verification on the transaction root in step 34, this implementation may further include as follows:

[171] verifying whether the output transaction root in the transaction root is the same as
30 an already-chained output transaction root and input transaction root. If the output transaction root in the transaction root is the same as the already-chained output transaction root or input transaction root, the first verification result is determined to be failed. If the output transaction root in the transaction root is different from the already-chained output transaction root and input transaction root, and has a valid signature, the first verification result is

determined to be successful.

[172] Specifically, the purpose of verifying the output transaction root is to avoid a "collision." The "collision" herein may be understood as follows: An input transaction root or an output transaction root is the same as the output transaction root, and both have the same (user) signature. Consequently, it cannot be determined that "money" corresponding to the transaction root is "money" the user has not "spent" or "money" the user has "spent."

[173] When the output transaction root is constructed, a random number is added to the post-transaction data. Therefore, the obtained output transaction root is unique. Accordingly, the output transaction root herein may be compared with the already-chained output transaction root and output transaction root to determine whether the output transaction root is the same as the already-chained output transaction root and input transaction root, and whether a collision occurs based on a comparison result.

[174] Another implementation of performing verification on the transaction root in step 34 may be as follows:

[175] Based on the foregoing implementation, whether an input transaction root in the transaction root is the same as the already-chained output transaction root and input transaction root is verified. The input transaction root is constructed according to pre-transaction data in the transaction data.

[176] In this implementation, the purpose of the input transaction root is to avoid a "dual payment." The "dual payment" herein may be understood as follows: The user pays "money" corresponding to the input transaction root to other two users. Details of an avoidance manner are similar to the output transaction root verification manner, and thus, are not described herein again.

[177] For the latter two implementations of step 34, the node may distinguish the input transaction root and the output transaction root in the following manner: An unspent transaction output (UTXO) of a transaction root is an input transaction root, and others are output transaction roots.

[178] An implementation of performing verification on the proof data in step 34 may be as follows:

[179] The consensus node performs verification on the proof data based on the transaction root by using a zero-knowledge proof algorithm. If the verification succeeds, it is determined that the proof data is correct.

[180] Step 36: Approve or reject the transaction request according to the first verification result and the second verification result.

[181] It should be noted that, if the first verification result and the second verification result both are successful, in response to the transaction request, the transaction root carrying a signature of the transaction initiator and the proof data is written on the blockchain; if the first verification result and/or the second verification result is failed, the transaction request is rejected.

[182] Accordingly, in this embodiment, at first, the one-way function is used to ensure that the transaction data is authentic and cannot be tampered with. Next, the proof data is constructed to prove validity of the information to be hidden without revealing any information to be hidden to a non-transaction participant. Compared with existing technologies, this embodiment can effectively prevent data privacy of the entire blockchain from being affected by a data leakage of any node, and improve data security of the blockchain.

[183] Embodiment 3

[184] FIG. 4 is a schematic flowchart of a data processing method, according to Embodiment 3 of the present specification. Referring to FIG. 4, the method may be performed by a node in a blockchain, and specifically includes the following steps:

[185] Step 42: A transaction node A (a transaction initiator) determines information to be hidden in transaction data.

[186] Step 44: The transaction node A obtains, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function, and constructs, according to the transaction root, proof data of validity of the information to be hidden.

[187] Step 46: The transaction node A sends the transaction root and the proof data to a transaction node B (a transaction receiver), so that the transaction receiver performs verification on the transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain.

[188] Step 48: The transaction node B performs verification on the transaction root according to the information to be hidden and the predetermined one-way function.

[189] Step 410: The transaction node B determines whether the verification succeeds.

[190] Step 412: If a verification result is successful, the transaction node B initiates a transaction request to write the proof data and the transaction root carrying the signatures of the transaction initiator and the transaction receiver on the blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and

approve or reject the transaction request according to a verification result.

[191] Step 414: A consensus node performs verification on the transaction root to obtain a first verification result, and performs, according to the transaction root, verification on the proof data to obtain a second verification result.

5 [192] An implementation of step 414 may be as follows:

[193] determining an address associated with an output transaction root in the transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data; and if determining that a quantity of obtained addresses is greater than or equal to a predetermined threshold, verifying whether the transaction root carries the signatures corresponding to the transaction initiator and the transaction receiver. The
10 predetermined threshold may be specifically 2.

[194] Understandably, each output transaction root has a corresponding address, and a node of a transaction participant is determined based on the address; next whether all the nodes of the transaction participants sign on the transaction root will be verified. If the
15 transaction root is signed and valid, the first verification result is successful.

[195] Step 416: Approve or reject the transaction request according to the first verification result and the second verification result.

[196] Accordingly, in this embodiment, at first, the one-way function is used to ensure that the transaction data is authentic and cannot be tampered with. Next, the proof data is
20 constructed to prove validity of the information to be hidden without revealing any information to be hidden to a non-transaction participant. Compared with existing technologies, this embodiment can effectively prevent data privacy of the entire blockchain from being affected by a data leakage of any node, and improve data security of the blockchain.

25 [197] Embodiment 4

[198] FIG. 5 is a schematic flowchart of a data processing method, according to Embodiment 4 of the present specification. Referring to FIG. 5, the method may be performed by a node in a blockchain, and specifically includes the following steps:

[199] Step 52: A transaction node A (a transaction initiator) determines information to be
30 hidden in transaction data.

[200] Step 54: The transaction node A obtains, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function, and constructs, according to the transaction root, proof data of validity of the information to be hidden.

[201] Step 56: The transaction node A sends the transaction root and the proof data to a transaction node B (a transaction receiver), so that the transaction receiver performs verification on the transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain.

[202] Step 58: The transaction node B performs verification on the transaction root according to the information to be hidden and the predetermined one-way function.

[203] Step 510: The transaction node B determines whether the verification succeeds.

[204] Step 512: If a verification result is successful, the transaction node B signs on the transaction root, and sends chaining-allowed indication information to the transaction node A.

[205] Step 514: The transaction node A initiates a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on the blockchain.

[206] Step 516: A consensus node performs verification on the transaction root to obtain a first verification result, and performs, according to the transaction root, verification on the proof data to obtain a second verification result.

[207] Step 518: Approve or reject the transaction request according to the first verification result and the second verification result.

[208] Accordingly, in this embodiment, first, the one-way function is used to ensure that the transaction data is authentic and cannot be tampered with, and then the proof data is constructed to prove validity of the information to be hidden without revealing any information to be hidden to a non-transaction participant. Compared with existing technologies, this embodiment can effectively prevent data privacy of the entire blockchain from being affected by a data leakage of any node, and improve data security of the blockchain.

[209] Embodiment 5

[210] With reference to FIG. 6a and FIG. 6b, the present specification is described below in detail by using an example, and may specifically include the following steps:

[211] Step S62. Referring to FIG. 6a, before a transaction is chained, a transaction in a blockchain is organized into a hierarchical one-way function structure with a bottom layer being a plaintext of the transaction, and a transaction initiator constructs a tree shape by using a one-way function, to eventually form a transaction root. The plaintext may be used as information to be hidden.

[212] Understandably, because all transaction content is organized by using the one-way function, chaining the transaction root does not leak any data privacy.

[213] Certainly, a transaction organization format may alternatively be any non-tree-shaped one-way function form.

5 [214] Step S64. Referring to FIG. 6b, if a particular function f needs to be calculated by using transaction data, f requires input of data 2 of a transaction 1 (whose transaction root is root), data 3' of a transaction 2 (whose transaction root is root'), and data 1'' of a transaction 3 (whose transaction root is root'') that are used as hidden information. That is,

[215] operation result = $f(\text{data 2, data 3', data 1''})$

10 [216] root = transaction root (data 2)

[217] root' = transaction root (data 3') and

[218] root'' = transaction root (data 1'')

[219] the transaction initiator proof may prove that the above stated is correct by performing a zero-knowledge proof, without leaking the data 2, the data 3', or the data 1''.

15 [220] It should be noted that, the particular function f is a function customized based on the hidden information that needs to be proved. A function type is not limited as long as the proof can be implemented. For example, pre-transaction balances and post-transaction balances of two parties of the transaction are used as inputs to calculate a difference between the two; or pre-transaction and post-transaction balances of the transaction initiator are used

20 as inputs to perform calculation to determine that a difference between the pre-transaction and post-transaction balances of the transaction initiator does not exceed m .

[221] Step S66. After receiving the proof, a receiver (a transaction receiver or a non-transaction-participant node in the blockchain) verifies correctness of the proof by performing a zero-knowledge proof to confirm correctness of an operation result.

25 [222] The receiver knows only the operation result. If the operation function does not leak information other than the operation result, the receiver does not know input data information of the function.

[223] Accordingly, in this embodiment, one party may prove to another party that already-chained transaction content may be used to calculate the particular function, but

30 operation of the function does not leak data privacy of any party. For the blockchain, the one-way function ensures that data content is authentic and cannot be tampered with, and the zero-knowledge proof ensures that a data operation process enables two parties to implement an agreed particular function, and the data truly belongs to an already-chained transaction.

[224] For brief description, the foregoing method embodiments are represented as a series

of actions. However, persons skilled in the art should appreciate that the present invention is not limited to the described order of the actions, because according to the present invention, some steps may be performed in other orders or simultaneously. In addition, persons skilled in the art should also appreciate that the embodiments described in the present disclosure are
5 exemplary embodiments, and an action involved is not necessarily mandatory to the embodiments of the present invention.

[225] Embodiment 6

[226] FIG. 7 is a schematic structural diagram of a data processing apparatus, according to Embodiment 6 of the present specification. Referring to FIG. 7, the apparatus may
10 specifically include: a determining unit 71, a processing unit 72, and an initiation unit 73.

[227] The determining unit 71 is configured to determine information to be hidden in transaction data.

[228] The processing unit 72 is configured to obtain, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the
15 predetermined one-way function, and construct, according to the transaction root, proof data of validity of the information to be hidden.

[229] The initiation unit 73 is configured to initiate a transaction request to write the proof data and the transaction root carrying a signature of the transaction initiator on a blockchain, for a node in the blockchain to perform consensus verification on the transaction
20 root and the proof data, and approve or reject the transaction request according to a verification result.

[230] In this embodiment, the determining unit 71 is further configured to determine pre-transaction data and post-transaction data in the transaction data, where

[231] correspondingly, the processing unit 72 is specifically configured to obtain, by
25 using the pre-transaction data as an input of a first predetermined one-way function, an input transaction root that is output by the first predetermined one-way function; obtain, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function; and use the obtained input transaction root and output transaction root as to-be-chained transaction roots.

[232] The processing unit 72 is specifically configured to construct a bottom layer of a predetermined one-way function tree according to the post-transaction data and a random
30 number, and determine an output transaction root corresponding to a top layer of the predetermined one-way function tree.

[233] The processing unit 72 is specifically configured to perform a predetermined

logical operation by using the transaction root and the information to be hidden as inputs; and perform a zero-knowledge proof on the logical operation by using the information to be hidden as a protected object to obtain the proof data.

[234] Embodiment 7

5 [235] FIG. 8 is a schematic structural diagram of a data processing apparatus, according to Embodiment 7 of the present specification. Referring to FIG. 8, the apparatus may specifically include: a receiving unit 81, a verification unit 82, and a response unit 83.

[236] The receiving unit 81 is configured to receive a transaction request from a transaction initiator, where the transaction request is used to request to write a transaction
10 root and proof data on a blockchain. The transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data.

[237] The verification unit 82 is configured to perform verification on the transaction root, to obtain a first verification result, and perform, according to the transaction root, verification
15 on the proof data to obtain a second verification result.

[238] The response unit 83 is configured to approve or reject the transaction request according to the first verification result and the second verification result.

[239] An operation principle of the verification unit 82 may be:

[240] verifying whether the transaction root has a valid signature, which is specifically:
20 determining an address associated with an output transaction root in the transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data; and verifying, if determining that a quantity of obtained addresses is 1, whether the transaction root has a signature corresponding to the transaction initiator;

[241] verifying whether an output transaction root in the transaction root is the same as an
25 already-chained output transaction root and input transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data; and

[242] verifying whether an input transaction root in the transaction root is the same as an already-chained output transaction root and input transaction root, where the input transaction root is constructed according to pre-transaction data in the transaction data.

30 [243] An operation principle of the response unit 83 may be:

[244] if the first verification result and the second verification result both are successful, writing, in response to the transaction request, the transaction root carrying a signature of the transaction initiator and the proof data on the blockchain; or if the first verification result and/or the second verification result is failed, rejecting the transaction request.

[245] Embodiment 8

[246] FIG. 9 is a schematic structural diagram of a data processing apparatus, according to Embodiment 8 of the present specification. Referring to FIG. 9, the apparatus may specifically include: a determining unit 91, a processing unit 92, and a sending unit 93.

5 [247] The determining unit 91 is configured to determine, by a transaction initiator, information to be hidden in transaction data.

[248] The processing unit 92 is configured to obtain, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function, and construct, according to the transaction root, proof data
10 of validity of the information to be hidden.

[249] The sending unit 93 is configured to send the transaction root and the proof data to a transaction receiver, where the transaction receiver performs verification on the transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction
15 receiver on a blockchain.

[250] Embodiment 9

[251] FIG. 10 is a schematic structural diagram of a data processing apparatus, according to Embodiment 9 of the present specification. Referring to FIG. 10, the apparatus may specifically include: a receiving unit 101, a verification unit 102, and an initiation unit 103.

20 [252] The receiving unit 101 is configured to receive a transaction root and proof data from a transaction sender, where the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data.

[253] The verification unit 102 is configured to perform verification on the transaction
25 root according to the information to be hidden and the predetermined one-way function.

[254] The initiation unit 103 is configured to: if a verification result is success, initiate a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or
30 reject the transaction request according to a verification result.

[255] Embodiment 10

[256] FIG. 11 is a schematic structural diagram of a data processing apparatus, according to Embodiment 10 of the present specification. Referring to FIG. 11, the apparatus may specifically include: a receiving unit 111, a verification unit 112, and a response unit 113.

[257] The receiving unit 111 is configured to receive a transaction request from a transaction receiver, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data initiated by a transaction initiator as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data.

[258] The verification unit 112 is configured to perform verification on the transaction root to obtain a first verification result, and perform, according to the transaction root, verification on the proof data to obtain a second verification result.

[259] The response unit 113 is configured to approve or reject the transaction request according to the first verification result and the second verification result.

[260] The verification unit 112 is specifically configured to determine an address associated with an output transaction root in the transaction root, where the output transaction root is constructed according to post-transaction data in the transaction data; and if determining that a quantity of obtained addresses is greater than or equal to a predetermined threshold, verify whether the transaction root carries signatures corresponding to the transaction initiator and the transaction receiver.

[261] The apparatus embodiments are basically similar to the method embodiments, and therefore are described briefly. For related parts, reference may be made to some descriptions in the method embodiments.

[262] It should be noted that among the components of the apparatus of the present invention, the components are logically divided according to the functions to be implemented by the components, but the present invention is not limited thereto, and the components may be redivided or combined as required.

[263] Embodiment 11

[264] FIG. 12 is a schematic structural diagram of an electronic device, according to Embodiment 11 of the present specification. Referring to FIG. 12, the electronic device includes a processor, an internal bus, a network interface, an internal memory, and a non-volatile memory, and certainly may further include other hardware necessary to services. The processor reads a corresponding computer program from the non-volatile memory, loads the computer program to the internal memory, and then runs the computer program, to logically form a data processing apparatus. Certainly, in addition to the software implementation, the present specification does not exclude other implementations, such as a logical device or a combination of hardware and software. That is, an entity for performing the following processing procedure is not limited to the logical units, but may be hardware or

a logical device.

[265] The network interface, the processor, and the memory may be connected to each other by using a bus system. The bus may be an industry standard architecture (ISA) bus, a peripheral component interconnect (PCI) bus, an extended industry standard architecture (EISA) bus, or the like. The bus may be classified into an address bus, a data bus, a control bus, and the like. For ease of representation, the bus is represented only by a double-sided arrow in FIG. 12, but it does not mean that there is only one bus or one type of bus.

[266] The memory is configured to store a program. Specifically, the program may include program code, and the program code includes a computer operation instruction. The memory may include a read-only memory (ROM) and a random access memory (RAM), and provides instructions and data to the processor. The memory may include a high-speed RAM, and may also include a non-volatile memory, such as at least one disk storage.

[267] The processor is configured to execute the program stored in the memory, and specifically configured to:

[268] determine information to be hidden in transaction data;

[269] obtain, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function, and construct, according to the transaction root, proof data of validity of the information to be hidden; and

[270] initiate a transaction request to write the proof data and the transaction root carrying a signature of the transaction initiator on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approves or reject the transaction request according to a verification result.

[271] The processor is further configured to:

[272] receive a transaction request from a transaction initiator, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[273] perform verification on the transaction root, to obtain a first verification result, and perform, according to the transaction root, verification on the proof data to obtain a second verification result; and

[274] approve or reject the transaction request according to the first verification result and the second verification result.

[275] The processor is further configured to:

[276] determine information to be hidden in transaction data;

[277] obtain, by using the transaction data as an input of a predetermined one-way function, a transaction root that is output by the predetermined one-way function, and construct, according to the transaction root, proof data of validity of the information to be

5 hidden; and

[278] send the transaction root and the proof data to a transaction receiver, where the transaction receiver performs verification on the transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain.

10 [279] The processor is further configured to:

[280] receive a transaction root and proof data from a transaction sender, where the transaction root is obtained by using transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[281] perform verification on the transaction root according to the information to be

15 hidden and the predetermined one-way function; and

[282] if a verification result is successful, initiate a transaction request to write the proof data and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction

20 request according to a verification result.

[283] The processor is further configured to:

[284] receive a transaction request from a transaction receiver, where the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is obtained by using transaction data initiated by a transaction initiator as an

25 input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

[285] perform verification on the transaction root to obtain a first verification result, and perform, according to the transaction root, verification on the proof data to obtain a second verification result; and

30 [286] approve or reject the transaction request according to the first verification result and the second verification result.

[287] The methods executed by the data processing apparatuses or a masternode disclosed in the embodiments shown in FIG. 2 to FIG. 10 of the present specification may be applied to the processor, or implemented by the processor. The processor may be an

integrated circuit chip and has a signal processing capability. In an implementation process, the steps of the foregoing method may be implemented by using a hardware integrated logic circuit in the processor or instructions in a form of software. The processor may be a general purpose processor, including a central processing unit (CPU), a network processor (NP), and the like; or may be a digital signal processor (DSP), an application-specific integrated circuit (ASIC), a field-programmable gate array (FPGA), or another programmable logic device, discrete gate or transistor logic device, or discrete hardware component. The processor may implement or execute methods, steps, and logical block diagrams disclosed in the embodiments of the present specification. The general purpose processor may be a microprocessor, or the processor may be any conventional processor or the like. The steps in the methods disclosed in the embodiments of the present specification may be directly implemented by a hardware decoding processor, or may be implemented by a combination of hardware and software modules in a decoding processor. The software module may be located in a mature storage medium in the art, such as a RAM, a flash memory, a read-only memory (ROM), a programmable ROM, an electrically erasable programmable memory, or a register. The storage medium is located in the memory. The processor reads information in the memory, and completes the steps of the method by using hardware thereof.

[288] The data processing apparatus may further execute the methods in FIG. 2 to FIG. 5, and implement the method executed by the master node.

[289] Embodiment 12

[290] Based on the same invention idea, this embodiment of the present specification further provides a computer readable storage medium. The computer readable storage medium stores one or more programs, and when the one or more programs are executed by an electronic device including a plurality of application programs, the electronic device is enabled to perform the data processing methods provided in Embodiments 1 to 5.

[291] The embodiments in this specification are all described in a progressive manner, for same or similar parts in the embodiments, refer to these embodiments, and descriptions of each embodiment focus on a difference from other embodiments. Especially, a system embodiment is basically similar to a method embodiment, and therefore is described briefly; for related parts, reference may be made to partial descriptions in the method embodiment.

[292] Embodiments of this specification are described above. Other embodiments fall within the scope of the appended claims. In some embodiments, the actions or steps recorded in the claims may be performed in sequences different from those in the embodiments and an expected result may still be achieved. In addition, the processes depicted in the

accompanying drawings do not necessarily require specific sequences or consecutive sequences to achieve an expected result. In some implementations, multitasking and parallel processing may be feasible or beneficial.

[293] 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 use a form of hardware only embodiments, software only embodiments, or embodiments with a combination of software and hardware. Moreover, the present invention may use a form of a computer program product that is implemented on one or more computer-usable storage media (including but not limited to a disk memory, a CD-ROM, an optical memory, and the like) that include computer-usable program code.

[294] The present invention is described with reference to the flowchart and/or block diagram of the method, device (system), and computer program product of the embodiments of the present invention. It should be understood that computer program instructions can implement each procedure and/or block in the flowcharts and/or block diagrams and a combination of procedures and/or blocks in the flowcharts and/or block diagrams. These computer program instructions may be provided to a general-purpose computer, a special-purpose computer, an embedded processor, or a processor of another programmable data processing device to generate a machine, so that an apparatus configured to implement functions specified in one or more procedures in the flowcharts and/or one or more blocks in the block diagrams is generated by using instructions executed by the general-purpose computer or the processor of another programmable data processing device.

[295] These computer program instructions may also be stored in a computer readable memory that can guide a computer or another programmable data processing device to work in a specific manner, so that the instructions stored in the computer readable memory generate a product including an instruction apparatus, where the instruction apparatus implements functions specified in one or more procedures in the flowcharts and/or one or more blocks in the block diagrams.

[296] These computer program instructions may also be loaded into a computer or another programmable data processing device, so that a series of operation steps are performed on the computer or another programmable data processing device to generate processing implemented by a computer, and instructions executed on the computer or another programmable data processing device provide steps for implementing functions specified in one or more procedures in the flowcharts and/or one or more blocks in the block diagrams.

[297] In a typical configuration, the computer device includes one or more processors

(CPUs), an input/output interface, a network interface, and a memory.

[298] The memory may include a form such as a persistent memory, a random-access memory (RAM) and/or a non-volatile memory of computer readable media, for example, a read-only memory (ROM) or a flash memory (RAM). The memory is an example of the computer readable medium.

[299] The computer readable medium includes a persistent medium and a non-persistent medium, a removable medium and a non-removable medium, which may implement storage of information by using any method or technology. The information may be a computer readable instruction, a data structure, a module of a program or other data. Examples of computer storage media include but are not limited to a phase change memory (PRAM), a static random access memory (SRAM), a dynamic random access memory (DRAM), other type of random access memory (RAM), a read-only memory (ROM), an electrically erasable programmable read-only memory (EEPROM), a flash memory or other memory technology, a compact disc read-only memory (CD-ROM), a digital versatile disc (DVD) or other optical storage, a cassette magnetic tape, tape and disk storage or other magnetic storage device or any other non-transmission media that may be configured to store information that a computing device can access. Based on the definition in the present invention, the computer-readable medium does not include transitory computer readable media (transitory media), such as a modulated data signal and a carrier.

[300] It should also be noted that the terms “include”, “comprise” and any other variants mean to cover the non-exclusive inclusion. Thereby, the process, method, article, or device which include a series of elements not only include those elements, but also include other elements which are not clearly listed, or include the inherent elements of the process, method, article and device. Without further limitation, the element defined by a phrase “include one” does not exclude other same elements in the process, method, article or device which include the element.

[301] A person skilled in the art should understand that the embodiments in the present invention may be provided as a method, a system, or a computer program product. Therefore, the present invention may use a form of hardware only embodiments, software only embodiments, or embodiments with a combination of software and hardware. Moreover, the present invention may use a form of a computer program product that is implemented on one or more computer-usable storage media (including but not limited to a disk memory, a CD-ROM, an optical memory, and the like) that include computer-usable program code.

[302] The foregoing descriptions are merely embodiments in the present invention and

are not intended to limit the present invention. For a person skilled in the art, various modifications and variations can be made to the present invention. Any modification, equivalent replacement, or improvement made within the spirit and principle of the present invention shall subject to the claims of the present invention.

CLAIMS

What is claimed is:

1. A data processing method, comprising:
 - determining, by a transaction initiator, information to be hidden in transaction data;
 - determining pre-transaction data and post-transaction data in the transaction data;
 - obtaining, by using the pre-transaction data as an input of a first predetermined one-way function, an input transaction root that is output by the first predetermined one-way function;
 - obtaining, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function;
 - using the input transaction root and the output transaction root as transaction roots for adding to a blockchain;
 - constructing, according to the input transaction root and the output transaction root, proof data of validity of the information to be hidden; and
 - initiating a transaction request to write the proof data and the input transaction root and the output transaction root carrying a signature of the transaction initiator on the blockchain, for a plurality of nodes in the blockchain to perform consensus verification on the input transaction root, the output transaction root, and the proof data, and approve or reject the transaction request according to a verification result.
2. The method according to claim 1, wherein the obtaining, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function comprises:
 - constructing, according to the post-transaction data and a random number, a bottom layer of a predetermined one-way function tree, and
 - determining the output transaction root corresponding to a top layer of the predetermined one-way function tree.
3. The method according to claim 1, wherein the constructing, according to the input transaction root and the output transaction root, proof data of validity of the information to be hidden comprises:
 - performing, by using the input transaction root, the output transaction root, and the information to be hidden as inputs, a predetermined logical operation; and

performing, by using the information to be hidden as a protected object, a zero-knowledge proof on the logical operation to obtain the proof data.

4. A data processing method, comprising:

receiving a transaction request from a transaction initiator, wherein the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is based on transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

performing verification on the transaction root, to obtain a first verification result, by verifying whether an output transaction root and an input transaction root in the transaction root is the same as a stored output transaction root and a stored input transaction root on the blockchain, wherein the output transaction root is based on post-transaction data in the transaction data as an input of a first predetermined one-way function, and wherein the input transaction root is based on pre-transaction data in the transaction data as an input of a second predetermined one-way function;

performing, according to the transaction root, verification on the proof data to obtain a second verification result; and

approving or rejecting the transaction request according to the first verification result and the second verification result.

5. The method according to claim 4, wherein the performing verification on the transaction root comprises:

verifying whether the transaction root has a valid signature.

6. The method according to claim 5, wherein the verifying whether the transaction root has a valid signature comprises:

determining an address associated with the output transaction root in the transaction root; and

verifying, if determining that a quantity of obtained addresses is 1, whether the transaction root has a signature corresponding to the transaction initiator.

7. The method according to claim 4, wherein the approving or rejecting the transaction request according to the first verification result and the second verification result comprises:

if the first verification result and the second verification result both are successful, writing,

in response to the transaction request, the transaction root carrying a signature of the transaction initiator and the proof data on the blockchain; or

if the first verification result and/or the second verification result is failed, rejecting the transaction request.

8. A data processing method, comprising:

determining, by a transaction initiator, information to be hidden in transaction data;

determining pre-transaction data and post-transaction data in the transaction data;

obtaining, by using the pre-transaction data as an input of a first predetermined one-way function, an input transaction root that is output by the first predetermined one-way function;

obtaining, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function;

using the input transaction root and the output transaction root as transaction roots for adding to a blockchain;

constructing, according to the input transaction root and the output transaction root, proof data of validity of the information to be hidden; and

sending the input transaction root, the output transaction root, and the proof data to a transaction receiver, wherein the transaction receiver performs verification on the input transaction root and the output transaction root, and initiates, after the verification succeeds, a transaction request to write the proof data, the input transaction root, and the output transaction root carrying signatures of the transaction initiator and the transaction receiver on the blockchain.

9. A data processing method, comprising:

receiving, by a transaction receiver, a transaction root and proof data from a transaction sender, wherein the transaction root includes an input transaction root and an output transaction root, wherein the input transaction root is obtained by using the pre-transaction data as an input of a first predetermined one-way function, and the output transaction root is obtained by using the post-transaction data as an input of a second predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

performing verification on the transaction root according to the information to be hidden and the first and second predetermined one-way functions; and

if a verification result is successful, initiating a transaction request to write the proof data

and the transaction root carrying signatures of the transaction initiator and the transaction receiver on a blockchain, for a node in the blockchain to perform consensus verification on the transaction root and the proof data, and approve or reject the transaction request according to a verification result.

10. A data processing method, comprising:

receiving a transaction request from a transaction receiver, wherein: the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root includes an input transaction root and an output transaction root, wherein the input transaction root is obtained by using the pre-transaction data as an input of a first predetermined one-way function, and the output transaction root is obtained by using the post-transaction data as an input of a second predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

performing verification on the transaction root to obtain a first verification result;

performing, according to the transaction root, verification on the proof data to obtain a second verification result; and

approving or rejecting the transaction request according to the first verification result and the second verification result.

11. The method according to claim 10, wherein the performing verification on the transaction root comprises:

determining an address associated with the output transaction root in the transaction root; and

if determining that a quantity of obtained addresses is greater than or equal to a predetermined threshold, verifying whether the transaction root carries signatures corresponding to the transaction initiator and the transaction receiver.

12. An apparatus comprising:

one or more processors; and

a memory storing instructions executable by the one or more processors to cause the one or more processors to perform operations comprising:

determining, by a transaction initiator, information to be hidden in transaction data;

determining pre-transaction data and post-transaction data in the transaction data;

obtaining, by using the pre-transaction data as an input of a first predetermined one-way

function, an input transaction root that is output by the first predetermined one-way function;

obtaining, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function;

using the input transaction root and the output transaction root as transaction roots for adding to a blockchain;

constructing, according to the input transaction root and the output transaction root, proof data of validity of the information to be hidden; and

initiating a transaction request to write the proof data and the input transaction root and the output transaction root carrying a signature of the transaction initiator on the blockchain, for a plurality of nodes in the blockchain to perform consensus verification on the input transaction root, the output transaction root, and the proof data, and approve or reject the transaction request according to a verification result.

13. The method according to claim 12, wherein the obtaining, by using the post-transaction data as an input of a second predetermined one-way function, an output transaction root that is output by the second predetermined one-way function comprises:

constructing, according to the post-transaction data and a random number, a bottom layer of a predetermined one-way function tree, and

determining the output transaction root corresponding to a top layer of the predetermined one-way function tree.

14. The method according to claim 12, wherein the constructing, according to the input transaction root and the output transaction root, proof data of validity of the information to be hidden comprises:

performing, by using the input transaction root, the output transaction root, and the information to be hidden as inputs, a predetermined logical operation; and

performing, by using the information to be hidden as a protected object, a zero-knowledge proof on the logical operation to obtain the proof data.

15. An apparatus comprising:

one or more processors; and

a memory storing instructions executable by the one or more processors to cause the one

or more processors to perform operations comprising:

receiving a transaction request from a transaction initiator, wherein the transaction request is used to request to write a transaction root and proof data on a blockchain, the transaction root is based on transaction data as an input of a predetermined one-way function, and the proof data is used to prove information to be hidden in the transaction data;

performing verification on the transaction root, to obtain a first verification result, by verifying whether an output transaction root and an input transaction root in the transaction root is the same as a stored output transaction root and a stored input transaction root on the blockchain, wherein the output transaction root is based on post-transaction data in the transaction data as an input of a first predetermined one-way function, and wherein the input transaction root is based on pre-transaction data in the transaction data as an input of a second predetermined one-way function;

performing, according to the transaction root, verification on the proof data to obtain a second verification result; and

approving or rejecting the transaction request according to the first verification result and the second verification result.

16. The method according to claim 15, wherein the performing verification on the transaction root comprises:

verifying whether the transaction root has a valid signature.

17. The method according to claim 16, wherein the verifying whether the transaction root has a valid signature comprises:

determining an address associated with the output transaction root in the transaction root; and

verifying, if determining that a quantity of obtained addresses is 1, whether the transaction root has a signature corresponding to the transaction initiator.

18. The method according to claim 15, wherein the approving or rejecting the transaction request according to the first verification result and the second verification result comprises:

if the first verification result and the second verification result both are successful, writing, in response to the transaction request, the transaction root carrying a signature of the transaction initiator and the proof data on the blockchain; or

if the first verification result and/or the second verification result is failed, rejecting the

transaction request.