

BLOCKCHAIN-BASED DATA VERIFICATION METHOD AND APPARATUS, AND ELECTRONIC DEVICE

TECHNICAL FIELD

[0001] One or more implementations of the present specification relate to the
5 blockchain field, and in particular, to data verification methods and apparatuses, and
electronic devices.

BACKGROUND

[0002] A blockchain technology, also referred to as a distributed ledger technology,
is a new technology in which a plurality of computing devices participate in
10 "accounting" to maintain a complete distributed database. The blockchain technology
has been widely used in many fields because of its features such as decentralization,
openness and transparency, and participation of each computing device in recording
data in a database, and fast data synchronization among computing devices.

SUMMARY

15 [0003] The present specification provides a blockchain-based data verification
method, including: obtaining a data identifier of target data published by a data
provider in a blockchain, where the data identifier indicates a storage location of the
target data in the blockchain; determining whether the target data has been stored in
the blockchain based on the data identifier; if the target data has been stored in the
20 blockchain, obtaining the target data from the storage location indicated by the data
identifier, where the target data stored in the blockchain carries verification
information used to verify authenticity of the target data; and verifying authenticity of
the obtained target data based on the verification information, and outputting a
verification result to a data verifier.

25 [0004] Optionally, the target data published by the data provider carries a graphic
code generated based on the data identifier; and obtaining a data identifier of target
data published by a data provider in a blockchain includes: scanning and parsing the
graphic code to obtain the data identifier of the target data in the blockchain.

[0005] Optionally, the data identifier is a transaction hash value of a transaction of the target data; and determining whether the target data has been stored in the blockchain based on the data identifier includes: performing Simplified Payment Verification (SPV) for the transaction based on a transaction hash value of the transaction of the target data, to determine whether the transaction has been stored in the blockchain.

[0006] Optionally, verifying authenticity of the obtained target data based on the verification information includes: executing an installed verification program to verify authenticity of the obtained target data based on the verification information; or invoking a smart contract published in the blockchain, executing a verification program defined in the smart contract, and verifying authenticity of the obtained target data based on the verification information.

[0007] Optionally, the target data stored in the blockchain further includes auxiliary information related to the authenticity verification; and the verification information is a hash value obtained by performing a hash calculation based on original content of the target data and the auxiliary information; and verifying authenticity of the obtained target data based on the verification information includes: performing a hash calculation on the original content of the target data and the auxiliary information to obtain a hash value; determining whether the calculated hash value is the same as a hash value carried in the target data stored in the blockchain; and if yes, determining that the obtained target data passed the authenticity verification, or if not, determining that the obtained target data has failed to pass the authenticity verification.

[0008] Optionally, the auxiliary information includes any one of or a combination of at least two of the following: an identifier of a data uploader; an upload timestamp of the target data; or an upload location of the target data.

[0009] Optionally, the method includes the following steps: outputting the target data obtained from the storage location indicated by the data identifier to the data verifier, so that the data verifier compares the outputted target data with the target data published by the data provider.

[0010] The present specification further provides a blockchain-based data verification apparatus, including: a first acquisition module, configured to obtain a data identifier of target data published by a data provider in a blockchain, where the data identifier indicates a storage location of the target data in the blockchain; a determining module, configured to determine whether the target data has been stored

in the blockchain based on the data identifier; a second acquisition module, configured to: if the target data has been stored in the blockchain, obtain the target data from the storage location indicated by the data identifier, where the target data stored in the blockchain carries verification information used to verify authenticity of the target data; and a verification module, configured to: verify authenticity of the obtained target data based on the verification information, and output a verification result to a data verifier.

[0011] Optionally, the target data published by the data provider carries a graphic code generated based on the data identifier; and the first acquisition module is configured to: scan and parse the graphic code to obtain the data identifier of the target data in the blockchain.

[0012] Optionally, the data identifier is a transaction hash value of a transaction of the target data; and the determining module is configured to: perform SPV for the transaction based on a transaction hash value of the transaction of the target data, to determine whether the transaction has been stored in the blockchain.

[0013] Optionally, the verification module is configured to: execute an installed verification program to verify authenticity of the obtained target data based on the verification information; or invoke a smart contract published in the blockchain, execute a verification program defined in the smart contract, and verify authenticity of the obtained target data based on the verification information.

[0014] Optionally, the target data stored in the blockchain further includes auxiliary information related to the authenticity verification; and the verification information is a hash value obtained by performing a hash calculation based on original content of the target data and the auxiliary information; and the verification module is further configured to: perform a hash calculation on the original content of the target data and the auxiliary information to obtain a hash value; determine whether the calculated hash value is the same as a hash value carried in the target data stored in the blockchain; and if yes, determine that the obtained target data passed the authenticity verification, or if not, determine that the obtained target data has failed to pass the authenticity verification.

[0015] Optionally, the auxiliary information includes any one of or a combination of at least two of the following: an identifier of a data uploader; an upload timestamp of the target data; or an upload location of the target data.

[0016] Optionally, the second acquisition module is further configured to: output

the target data obtained from the storage location indicated by the data identifier to the data verifier, so that the data verifier compares the outputted target data with the target data published by the data provider.

[0017] The present specification further provides an electronic device, including: a processor; and a memory, configured to store machine executable instructions; where by reading and executing the machine executable instructions that are stored in the memory and that correspond to a control logic for blockchain-based data verification, the processor is enabled to: obtain a data identifier of target data published by a data provider in a blockchain, where the data identifier indicates a storage location of the target data in the blockchain; determine whether the target data has been stored in the blockchain based on the data identifier; if the target data has been stored in the blockchain, obtain the target data from the storage location indicated by the data identifier, where the target data stored in the blockchain carries verification information used to verify authenticity of the target data; and verify authenticity of the obtained target data based on the verification information, and output a verification result to a data verifier.

[0018] According to the previously described technical solution, on one hand, it is determined, based on the data identifier published by the data provider, whether the target data has been stored in the blockchain, and authenticity of the information published by the data provider can be verified, so that the data verifier can quickly identify a false data identifier that is published by the data provider and that does not correspond to any data storage record in the blockchain.

[0019] On the other hand, after it is determined that the target data has been stored in the blockchain, authenticity verification is performed on the target data in the blockchain based on the verification information carried in the target data stored in the blockchain, so that the data verifier can determine whether the content of the target data in the blockchain has been tampered with, and quickly identify abnormal data.

BRIEF DESCRIPTION OF DRAWINGS

[0020] FIG. 1 is a schematic diagram illustrating a blockchain-based data verification method, according to an example implementation;

[0021] FIG. 2 is a diagram illustrating a consortium blockchain, according to an example implementation;

[0022] FIG. 3 is a schematic structural diagram illustrating an electronic device, according to an example implementation; and

[0023] FIG. 4 is block diagram illustrating a blockchain-based data verification apparatus, according to an example implementation.

5

DESCRIPTION OF IMPLEMENTATIONS

[0024] As the level of social digitalization increases, how to connect the real world and the digital world has become a key concern when the Internet goes deeper into more fields of life and industries.

[0025] For example, in a typical scenario, some products are sold on an e-commerce website, and because the users can only see some pictures (digital information) on the website, it is necessary to convince consumers that the digital information corresponds to a product in the real world in actual applications.

[0026] In view of this, the present specification provides a technical solution to verify authenticity of the digital information stored in the distributed database of the blockchain.

[0027] During implementation, a standard data storage format can be defined for the target data, and the target data is uploaded by the uploading client to the blockchain node device accessible by the uploading client according to the defined data storage format. The defined data storage format can include verification information used to verify authenticity of the target data.

[0028] After receiving the target data uploaded by the uploading client, the node device can publish the target data in the blockchain, and store the target data in the distributed database of the blockchain after the consensus processing among the node devices in the blockchain.

[0029] After the target data is successfully stored in the distributed database of the blockchain, the node device in the blockchain can return a data identifier indicating the storage location of the target data in the blockchain to the data provider through the uploading client. For example, in actual applications, the data identifier can specifically be a transaction hash value of a transaction of the target data.

[0030] Further, after the target data is successfully stored in the distributed database of the blockchain, the data provider can publish the target data and the data identifier of the target data in the blockchain returned by the node device in the blockchain, for

example, through a website.

[0031] After viewing the target data published by the data provider and the data identifier of the target data in the blockchain, the data verifier can verify, through the verification client, authenticity of the target data published by the data provider.

5 **[0032]** First, the verification client can obtain the data identifier published by the data provider, and verify, based on the data identifier, whether the target data has been stored in the distributed database of the blockchain.

[0033] Further, if it is verified that the target data has been stored in the distributed database of the blockchain, the target data stored in the blockchain can be obtained
10 from the storage location indicated by the data identifier, and then authenticity verification is performed on the target data based on the verification information carried in the target data, and a verification result is output to the data verifier by the verification client.

[0034] According to the previously described technical solution, on one hand, it is
15 determined, based on the data identifier published by the data provider, whether the target data has been stored in the blockchain, and authenticity of the information published by the data provider can be verified, so that the data verifier can quickly identify a false data identifier that is published by the data provider and that does not correspond to any data storage record in the blockchain.

20 **[0035]** On the other hand, after it is determined that the target data has been stored in the blockchain, authenticity verification is performed on the target data in the blockchain based on the verification information carried in the target data stored in the blockchain, so that the data verifier can determine whether the content of the target data in the blockchain has been tampered with, and quickly identify abnormal data.

25 **[0036]** The following describes the present specification by using the implementations and with reference to the specific application scenarios.

[0037] Referring to FIG. 1, FIG. 1 illustrates a blockchain-based data verification method, according to an implementation of the present specification. The method is applied to a verification client in a blockchain and includes the following steps:

30 **[0038]** Step 102: Obtain a data identifier of target data published by a data provider in a blockchain, where the data identifier indicates a storage location of the target data in the blockchain.

[0039] Step 104: Determine whether the target data has been stored in the blockchain based on the data identifier.

[0040] Step 106: If the target data has been stored in the blockchain, obtain the target data from the storage location indicated by the data identifier, where the target data stored in the blockchain carries verification information used to verify authenticity of the target data.

5 **[0041]** Step 108: Verify authenticity of the obtained target data based on the verification information, and output a verification result to a data verifier.

[0042] The target data can specifically include any form of digital information that can be stored in a blockchain, that can be released through the Internet, and that corresponds to an object in the real world.

10 **[0043]** For example, in one scenario, the target data can specifically be a product quality inspection report issued by a brick-and-mortar quality inspection organization, and the merchant can release a picture of the product together with a picture of the product quality inspection report of the product when issuing the product through an electronic commerce platform.

15 **[0044]** The data identifier can specifically include any form of identification information indicating a storage location of the data stored in the blockchain.

[0045] For example, in an illustrated implementation, in a distributed database of a blockchain, data is typically recorded in a transaction, stored in a block in the blockchain, and the transaction stored in the block is typically organized into a Merkle tree based on the transaction hash value of the transaction. Therefore, to facilitate data retrieval, the data identifier can specifically be a transaction hash value of a transaction of data. The transaction of the data can be quickly located based on a transaction hash value. Of course, in actual applications, the data identifier can specifically be a transaction index of a transaction of the data.

25 **[0046]** The blockchain described in the present specification can specifically include any type of blockchain network.

[0047] For example, in an illustrated implementation, the blockchain can specifically be a consortium blockchain including a plurality of merchants, a plurality of quality inspection organizations, and a plurality of consumers. As a data uploader, 30 the quality inspection organization can access any node device in the consortium blockchain through the uploading client, and upload the product quality inspection report to the node device in the form of digital information. The node device initiates consensus processing for the product quality inspection report in the blockchain, and stores the product quality inspection report in the distributed database of the

blockchain after the consensus is passed. As a data provider, a merchant can release, through an electronic commerce platform, a product quality inspection report issued by a quality inspection organization. When viewing the product quality inspection report released by the merchant, the consumer can also initiate authenticity verification for the product quality inspection report by accessing any node device in the consortium blockchain through the verification client.

[0048] The following describes the technical solution of the present specification in detail based on an example in which the target data is a product quality inspection report, the blockchain is the previously described consortium blockchain including a plurality of merchants, a plurality of quality inspection organizations, and a plurality of consumers.

[0049] Referring to FIG. 2, FIG. 2 is a diagram illustrating a consortium blockchain, according to the present specification.

[0050] As shown in FIG. 2, a blockchain operator can pre-establish a consortium blockchain including a plurality of node devices. Merchants, product quality inspection organizations, and consumers can register in the consortium blockchain, and access the consortium blockchain as members of the consortium blockchain after obtaining legal identities in the consortium blockchain (for example, obtaining private key/public key pairs assigned by the consortium blockchain).

[0051] A specific process in which a merchant, a product quality inspection organization, and a consumer are registered in a consortium blockchain and join the consortium blockchain as members of the consortium blockchain is omitted in the present specification for simplicity.

[0052] Referring to FIG. 2, as a data uploader, the product quality inspection organization can access a node device in a consortium blockchain through an uploading client, and upload a product quality inspection report issued for a product entrusted by a merchant to the node device in the form of digital information.

[0053] As a data verifier, a consumer can access a node device in a consortium blockchain through a verification client. When viewing a product quality inspection report released by a merchant, the consumer can initiate authenticity verification for the product quality inspection report through the verification client.

[0054] It is worthwhile to note that the uploading client and the verification client each can be a web client (such as a blockchain browser) or a pre-installed application (APP), which is not specifically limited in the present specification.

[0055] In the present specification, the product quality inspection organization can define a standard data storage format for a product quality inspection report to be uploaded. After the product quality inspection organization issues a product quality inspection report for the product entrusted by the merchant, the product quality inspection report can be organized into a standard data storage format, signed based on the private key held by the product quality inspection organization, and then uploaded to the blockchain node device accessible by the uploading client.

[0056] The data storage format defined for the product quality inspection report can include three parts: original content of the data, verification information, and auxiliary information. The verification information is specifically used to verify authenticity of the data. The auxiliary information can be an auxiliary parameter related to the authenticity verification.

[0057] In an illustrated implementation, the verification information can be a hash value obtained by performing a hash calculation based on the original content of the product quality inspection report and the auxiliary information. The auxiliary information can include one of or a combination of at least two of the following: an identity of a data uploader, an upload timestamp, an upload location, etc.

[0058] In this case, the data storage format defined for the product quality inspection report can be shown in Table 1.

Number	Field name	Field content
1	Content field	Original content of data
2	Uploader identity field	Identity of an uploader
3	Upload time field	Upload timestamp of the data
4	Upload location field	Upload location of the data
5	Verification information field	A hash value calculated based on fields 1-4

Table 1

[0059] The specific form of the content field entry is not specifically limited in the present specification. For example, it can be in the form of a picture.

[0060] The uploader identity field entry can be any form of information indicating the identity of the uploader, for example, an identity card number, fingerprint information, or a digital certificate signature.

[0061] The upload time field entry should be based on the generally accepted world

time, so as to prevent ambiguity caused by inconsistency in understanding the same time between people in different time zones. For example, people in the Beijing time zone and people in the New York time zone may have inconsistent understanding of the same time.

5 **[0062]** The address description in the upload location field should not cause ambiguity. For example, there is "Zhongshan Road" in many cities. The most appropriate way is to use unambiguous address expression such as longitude and latitude coordinates, GeoHash, etc.

10 **[0063]** In the present specification, after receiving the product quality inspection report uploaded by the uploading client, the node device in the consortium blockchain can first verify the signature of the product quality inspection report based on the public key corresponding to the private key held by the product quality inspection organization. If the signature verification is passed, it indicates that the product quality inspection report is the information uploaded by the legitimate product quality
15 inspection organization. In this case, the target data can be published in the blockchain, and the product quality inspection report can be stored in the distributed database of the blockchain after the consensus processing among the node devices in the consortium blockchain is complete.

20 **[0064]** The consensus algorithm used in the consortium blockchain is not specifically limited in the present specification. For example, mainstream consensus algorithms suitable for use in a consortium blockchain, such as PBFT, can be used, or a related consensus algorithm can be independently developed by the operator of the consortium blockchain.

25 **[0065]** When the product quality inspection report is successfully stored in the distributed database of the blockchain, the node device in the blockchain can return a data identifier indicating the storage location of the product quality inspection report in the blockchain to the product quality inspection organization through the uploading client.

30 **[0066]** For example, in an illustrated implementation, the data identifier can specifically be a transaction hash value of a transaction of the product quality inspection report.

[0067] In the present specification, the product quality inspection organization can also send the generated product quality inspection report and the data identifier returned by the node device in the consortium blockchain to the merchant. After

receiving the product quality inspection report and the data identifier that are sent by the product quality inspection organization, the merchant can publish the product quality inspection report and the data identifier to a consumer through a website. For example, a picture of a product can be published to a consumer together with a picture of the product through an electronic merchant platform.

[0068] After viewing the product quality inspection report released by the merchant and the data identifier of the product quality inspection report in the blockchain, the consumer can verify authenticity of the product quality inspection report released by the merchant through the verification client.

[0069] In an illustrated implementation, after viewing the data identifier of the product quality inspection report in the blockchain that is published by the merchant, the consumer can manually enter the data identifier on the user interface provided by the verification client. The verification client can obtain the data identifier manually input by the user, and then initiate, based on the obtained data identifier, verification of the product quality inspection report.

[0070] In another illustrated implementation, when publishing the product quality inspection report, the merchant can add a graphic code (such as a two-dimensional code or a bar code) of the product quality inspection report to the released product quality inspection report.

[0071] For example, if the data identifier is a transaction hash value of a transaction of the product quality inspection report, the merchant can generate a corresponding graphic code by using a character string corresponding to the transaction hash value as coding information, and then attach the generated graphic code to the picture of the released product quality inspection report.

[0072] In this case, the consumer can scan and parse the graphic code by using the verification client to obtain the data identifier carried in the graphic code, and then initiate, based on the parsed data identifier, verification of the product quality inspection report.

[0073] In the present specification, after the verification client obtains the data identifier of the product quality inspection report in the consortium blockchain that is published by the merchant, it can be determined, based on the data identifier, whether the product quality inspection report has been stored in the distributed database of the consortium blockchain.

[0074] In an illustrated implementation, the data identifier is a transaction hash

value of the transaction of the product quality inspection report, and Simplified Payment Verification (SPV) can be performed on the transaction based on the transaction hash value, to determine whether the transaction has been stored in the distributed database of the consortium blockchain.

5 **[0075]** SPV is a protocol used to verify whether a transaction already exists in the distributed database of the blockchain. The following describes in detail a process of performing SPV on the transaction based on the transaction hash.

[0076] In actual applications, a block in a blockchain generally includes two parts: a block header and a block body (including a transaction). A transaction recorded in a block is generally organized into a Merkle tree in the form of a transaction hash value.

10 **[0077]** The detailed process of organizing a hash value of a transaction recorded in a Merkle tree is not specifically limited in the present specification. A person skilled in the art can implement the technical solution in the present specification by referring to related technologies.

15 **[0078]** When SPV is performed on a transaction based on a transaction hash value, a Merkle verification path of the transaction including a target block of the transaction in a Merkle tree can be obtained first.

[0079] The Merkle verification path is specifically a path formed by sibling nodes (that is, neighboring nodes) that are obtained by traversing the Merkle tree based on the transaction value of a transaction. When SPV is performed on a transaction, the Merkle verification path of the transaction can be used as a calculation parameter for inversely calculating a hash value corresponding to the root node of the Merkle tree at which the transaction is located.

20 **[0080]** It is worthwhile to note that the Merkle verification path can be manually submitted by the user, or can be proactively queried from the blockchain by the verification client.

[0081] For example, when the verification client proactively queries the Merkle verification path of the transaction in the Merkle tree including the target block of the transaction, the verification client can first locate the target block in which the transaction is located based on the transaction hash value of the transaction.

30 **[0082]** A process of locating a block in which a transaction is located based on a transaction hash value is omitted in the present specification for simplicity. For example, in the related art, a Bloom filter can be deployed to locate a block in which a hash value of a transaction is located. After locating the block in which the transaction

is located, the Merkle verification path of the transaction in the Merkle tree can be further found from the located Merkle tree of the block.

[0083] In the present specification, after obtaining the Merkle verification path of the transaction in the Merkle tree including the target block of the transaction, the
5 hash value of the block header of the target block (that is, the hash value of the root node of the Merkle tree of the target block) can be calculated based on the calculation procedure specified by the SPV protocol. Then, it can be determined whether the calculated hash value of the block header of the target block matches the hash value stored in the block header of the target block. If the calculated hash value of the block
10 header of the target block matches the hash value stored in the block header of the target block, it can be determined that the transaction is included in the target block; that is, the product quality inspection report has been successfully stored in the distributed database of the consortium blockchain. Alternatively, if the calculated hash value of the block header of the target block does not match the hash value stored in
15 the block header of the target block, it indicates that the transaction is not included in the target block; that is, the product quality inspection report is not successfully stored in the distributed database stored in the consortium blockchain.

[0084] In the present specification, when it is determined, through the previously described implementation process, that the product quality inspection report has been
20 stored in the distributed database of the consortium blockchain, the product quality inspection report can be further obtained from the storage location indicated by the data identifier.

[0085] For example, if the data identifier is still the transaction hash value of the transaction of the product quality inspection report, the corresponding transaction can
25 be located based on the transaction hash value, and then the product quality inspection report recorded in the transaction can be read from the located transaction.

[0086] After the product quality inspection report is obtained from the storage location indicated by the data identifier, the verification information carried in the standard storage format of the product quality inspection report can be further read,
30 and then authenticity of the product quality inspection report obtained from the distributed database of the consortium blockchain can be verified based on the verification information.

[0087] In the present specification, the authenticity verification is specifically to verify whether the original content recorded in the product quality inspection report is

tampered with.

[0088] For example, in some scenarios, after a product quality inspection report is issued by a product quality inspection organization, the product quality inspection report can be organized into a standard data storage format and then sent to a merchant for uploading. In this case, some less reputable merchants can maliciously modify certain fields in the product quality inspection report before uploading the report.

[0089] For another example, in other scenarios, the product quality inspection report can be illegally intercepted during transmission, and certain fields in the report can be maliciously modified.

[0090] By introducing the process for verifying authenticity of the product quality inspection report in the blockchain, consumers can quickly identify malicious modifications of the product quality inspection report listed above.

[0091] In the present specification, when the verification client verifies authenticity of the product quality inspection report obtained from the distributed database of the consortium blockchain based on the verification information carried in the product quality inspection report, the verification client can specifically execute a locally installed verification program, or can implement the verification by remotely invoking a smart contract deployed in the consortium blockchain.

[0092] In an illustrated implementation, a verification program can be installed on the verification client to verify authenticity of the product quality inspection report.

[0093] For example, the authenticity verification rule of the product quality inspection report can be released by the product quality inspection organization serving as a data uploader, and the developer of the verification client can develop corresponding program execution code (for example, a function corresponding to the execution logic of the verification rule) in the software environment of the verification client based on the verification rule published by the product quality inspection organization.

[0094] When the verification client determines that the product quality inspection report has been stored in the distributed database of the consortium blockchain, and obtains the product quality inspection report stored in the consortium blockchain, the verification client can read the verification information from the obtained product quality inspection report. For example, when the data storage format shown in Table 1 is used, the verification client can read the verification information from field 5 shown

in Table 1.

[0095] Further, the verification program installed locally can be executed to verify authenticity of the product quality inspection report obtained from the consortium blockchain based on the read verification information.

5 **[0096]** In an illustrated implementation, a smart contract can alternatively be pre-deployed in the consortium blockchain, and a verification procedure for verifying authenticity of the product quality inspection report is defined in the smart contract.

[0097] For example, an operator of a consortium blockchain can develop corresponding contract code based on an authenticity verification rule for the product
10 quality inspection report released by a product quality inspection organization as a data uploader, deploy a smart contract, and then publish the smart contract in the consortium blockchain. Then, a node device in the consortium blockchain can perform consensus processing on the smart contract, and store the smart contract in a distributed database of the consortium blockchain after the consensus processing is
15 complete, so that a client accessing the consortium blockchain can remotely invoke the smart contract.

[0098] When the verification client determines that the product quality inspection report has been stored in the distributed database of the consortium blockchain, and obtains the product quality inspection report stored in the consortium blockchain, the
20 verification client can read the verification information from the obtained product quality inspection report, construct a transaction based on the verification information, submit the transaction to the smart contract, initiate revocation of the smart contract, execute a verification program defined in the smart contract, and verify authenticity of the product quality inspection report obtained from the consortium blockchain based
25 on the read verification information.

[0099] In an illustrated implementation, for example, the verification information is a hash value obtained by performing a hash calculation based on the original content of the product quality inspection report and the auxiliary information carried in the data storage format of the product quality inspection report. In this case, the
30 verification program installed locally on the verification client is executed; or the smart contract is invoked remotely, and the verification program defined in the smart contract is executed. When authenticity verification is performed on the product quality inspection report, the hash value can be obtained through re-calculation based on the original content of the product quality inspection report and the auxiliary

information carried in the data storage format of the product quality inspection report obtained from the consortium blockchain.

[0100] Further, the calculated hash value can be matched with the hash value that is carried in the data storage format of the product quality inspection report as the verification information, to determine whether the calculated hash value is the same as the hash value that is carried in the data storage format of the product quality inspection report as the check information. If the two are the same, it indicates that the content carried in the product quality inspection report stored in the consortium blockchain is not tampered with, and the product quality inspection report passes the authenticity verification; of course, if the two are different, it indicates that the content carried in the product quality inspection report stored in the consortium blockchain has been tampered with, and the product quality inspection report does not pass the authenticity verification.

[0101] After the authenticity verification for the product quality inspection report is complete, the authenticity verification result can be output by the verification client to a consumer. Then, the consumer can determine whether to purchase the product by referring to the authenticity verification result output by the verification client for the product quality inspection report.

[0102] In the present specification, the product quality inspection report obtained from the consortium blockchain can also be output to a consumer through the verification client, so that the consumer can compare the stored product quality inspection report in the consortium blockchain with the product quality inspection report released by the merchant.

[0103] In this way, the consumer can visually check whether the information recorded in the product quality inspection report released by the merchant is consistent with the information recorded in the product quality inspection report uploaded by the product quality inspection organization.

[0104] In actual applications, the product quality inspection report can be output to the consumer immediately after being obtained from the consortium blockchain; or the product quality inspection report can be selectively outputted to a consumer after the authenticity verification for the product quality inspection report is complete and the authenticity verification result is output to the consumer by the verification client.

[0105] For example, when the authenticity verification result indicates that the product quality inspection report has passed the authenticity verification, the product

quality inspection report obtained from the consortium blockchain can be output and displayed to the consumer. On the contrary, if the product quality inspection report does not pass the authenticity verification, the product quality inspection report is no longer output to the consumer. In this way, it is ensured that the consumer can only
 5 view the product quality inspection report that has passed the authenticity verification.

[0106] It is worthwhile to note that the product quality inspection report in the above implementation is only an example of the target data. Clearly, in actual applications, the target data can alternatively be data in other forms other than the product quality inspection report. When the target data is data in another form other
 10 than a product quality inspection report, a person skilled in the art can implement the technical solution described in the present specification by referring to the implementation process described in the implementations. Details are omitted in the present specification for simplicity.

[0107] According to the previously described technical solution, on one hand, it is
 15 determined, based on the data identifier published by the data provider, whether the target data has been stored in the blockchain, and authenticity of the information published by the data provider can be verified, so that the data verifier can quickly identify a false data identifier that is published by the data provider and that does not correspond to any data storage record in the blockchain.

[0108] For example, if it is determined, based on the data identifier published by
 20 the merchant, that no product quality inspection report corresponding to the data identifier is stored in the blockchain, it indicates that the data identifier published by the merchant may be a false data identifier, that is, the product quality inspection report released by the merchant may not exist.

[0109] On the other hand, after it is determined that the target data has been stored
 25 in the blockchain, authenticity verification is performed on the target data in the blockchain based on the verification information carried in the target data stored in the blockchain, so that the data verifier can determine whether the content of the target data in the blockchain has been tampered with, and quickly identify abnormal data.

[0110] For example, if the product quality inspection report stored in the
 30 blockchain does not pass the authenticity verification, it indicates that the product quality inspection report can be illegally tampered with before or when the report is uploaded, so that the consumer can quickly identify the abnormality.

[0111] Corresponding to the previously described method implementations, the

present specification further provides an implementation of a blockchain-based data verification apparatus. The implementations of the blockchain-based data verification apparatus in the present specification can be applied to an electronic device. The apparatus implementations can be implemented by using software, hardware, or a combination of software and hardware. The software implementation is used as an example. As a logical apparatus, the apparatus is formed by reading the corresponding computer program instructions in the non-volatile memory by the processor of the electronic device into the memory for execution. In terms of hardware, FIG. 3 is a diagram illustrating a hardware structure of an electronic device in which a blockchain-based data verification apparatus is located. In addition to a processor, a memory, a network interface, and a non-volatile memory shown in FIG. 3, the electronic device can generally include other hardware based on other actual functions of the electronic device. Details are omitted here for simplicity.

[0112] FIG. 4 is a logical block diagram illustrating a blockchain-based data verification apparatus, according to an example implementation of the present specification.

[0113] Referring to FIG. 4, the blockchain-based data verification apparatus 40 can be applied to the electronic device shown in FIG. 3, and includes a first acquisition module 401, a determining module 402, a second acquisition module 403, and a verification module 404.

[0114] The first acquisition module 401 is configured to obtain a data identifier of target data published by a data provider in a blockchain, where the data identifier indicates a storage location of the target data in the blockchain.

[0115] The determining module 402 is configured to determine whether the target data has been stored in the blockchain based on the data identifier.

[0116] The second acquisition module 403 is configured to: if the target data has been stored in the blockchain, obtain the target data from the storage location indicated by the data identifier, where the target data stored in the blockchain carries verification information used to verify authenticity of the target data.

[0117] The verification module 404 is configured to: verify authenticity of the obtained target data based on the verification information, and output a verification result to a data verifier.

[0118] In this implementation, the target data published by the data provider carries a graphic code generated based on the data identifier; and the first acquisition module

401 is configured to: scan and parse the graphic code to obtain the data identifier of the target data in the blockchain.

[0119] In this implementation, the data identifier is a transaction hash value of a transaction of the target data; and the determining module 402 is configured to:
5 perform SPV for the transaction based on a transaction hash value of the transaction of the target data, to determine whether the transaction has been stored in the blockchain.

[0120] In this implementation, the verification module 404 is further configured to:
10 execute an installed verification program to verify authenticity of the obtained target data based on the verification information; or invoke a smart contract published in the blockchain, execute a verification program defined in the smart contract, and verify authenticity of the obtained target data based on the verification information.

[0121] In this implementation, the target data stored in the blockchain further includes auxiliary information related to the authenticity verification; and the
15 verification information is a hash value obtained by performing a hash calculation based on original content of the target data and the auxiliary information; and the verification module 404 is further configured to: perform a hash calculation on the original content of the target data and the auxiliary information to obtain a hash value; determine whether the calculated hash value is the same as a hash value carried in the
20 target data stored in the blockchain; and if yes, determine that the obtained target data passed the authenticity verification, or if not, determine that the obtained target data has failed to pass the authenticity verification.

[0122] In this implementation, the auxiliary information includes any one of or a combination of at least two of the following: an identifier of a data uploader; an
25 upload timestamp of the target data; or an upload location of the target data.

[0123] In this implementation, the second acquisition module 403 is further configured to: output the target data obtained from the storage location indicated by the data identifier to the data verifier, so that the data verifier compares the outputted target data with the target data published by the data provider.

30 [0124] For the detailed implementation process of the functions and purposes of the modules in the apparatus, references can be made to the implementation process of the corresponding steps in the method, and details are omitted here for simplicity.

[0125] Because the apparatus implementation basically corresponds to the method implementation, for the related parts, references can be made to the description of the

method implementation. The previously described apparatus implementation is merely an example, where the units described as separate parts can or does not have to be physically separate, and components displayed as units can or cannot be physical units, and can be located in one place or can be distributed on a plurality of network units. Based on the practical needs, some or all of these modules can be selected to implement the purpose of the present specification. A person of ordinary skill in the art can understand and implement the blockchain-based data processing apparatus without creative efforts.

[0126] The system, apparatus, module, or unit illustrated in the previously described implementations can be implemented by using a computer chip or an entity, or can be implemented by using a product with a certain function. A typical implementation device is a computer in the form of a personal computer, a laptop computer, a cellular phone, a camera phone, a smart phone, a personal digital assistant, a media player, a navigation device, an e-mail transceiver, a game console, a tablet computer, a wearable device, or any combination of at least two of these devices.

[0127] Corresponding to the previously described method implementation, the present specification further provides an implementation of an electronic device. The electronic device includes a processor and a memory configured to store a machine executable instruction, where the processor and memory are usually connected to each other through an internal bus. In other possible implementations, the electronic device can also include an external interface used to communicate with other devices or components.

[0128] In this implementation, by reading and executing the machine executable instructions that are stored in the memory and that correspond to a control logic for blockchain-based data verification, the processor is enabled to: obtain a data identifier of target data published by a data provider in a blockchain, where the data identifier indicates a storage location of the target data in the blockchain; determine whether the target data has been stored in the blockchain based on the data identifier; if the target data has been stored in the blockchain, obtain the target data from the storage location indicated by the data identifier, where the target data stored in the blockchain carries verification information used to verify authenticity of the target data; and verify authenticity of the obtained target data based on the verification information, and output a verification result to a data verifier.

[0129] In this implementation, the target data published by the data provider carries

a graphic code generated based on the data identifier; and by reading and executing the machine executable instructions that are stored in the memory and that correspond to a control logic for blockchain-based data verification, the processor is enabled to: scan and parse the graphic code to obtain the data identifier of the target data in the
5 blockchain.

[0130] In this implementation, the data identifier is a transaction hash value of a transaction of the target data; and by reading and executing the machine executable instructions that are stored in the memory and that correspond to a control logic for blockchain-based data verification, the processor is enabled to: perform SPV for the
10 transaction based on a transaction hash value of the transaction of the target data, to determine whether the transaction has been stored in the blockchain.

[0131] In this implementation, by reading and executing the machine executable instructions that are stored in the memory and that correspond to a control logic for blockchain-based data verification, the processor is enabled to: execute an installed
15 verification program to verify authenticity of the obtained target data based on the verification information; or invoke a smart contract published in the blockchain, execute a verification program defined in the smart contract, and verify authenticity of the obtained target data based on the verification information.

[0132] In this implementation, the target data stored in the blockchain further
20 includes auxiliary information related to the authenticity verification; and the verification information is a hash value obtained by performing a hash calculation based on original content of the target data and the auxiliary information; and by reading and executing the machine executable instructions that are stored in the memory and that correspond to a control logic for blockchain-based data verification,
25 the processor is enabled to: perform a hash calculation on the original content of the target data and the auxiliary information to obtain a hash value; determine whether the calculated hash value is the same as a hash value carried in the target data stored in the blockchain; and if yes, determine that the obtained target data passed the authenticity verification, or if not, determine that the obtained target data has failed to
30 pass the authenticity verification.

[0133] In this implementation, by reading and executing the machine executable instructions that are stored in the memory and that correspond to a control logic for blockchain-based data verification, the processor is enabled to: output the target data obtained from the storage location indicated by the data identifier to the data verifier,

so that the data verifier compares the outputted target data with the target data published by the data provider.

[0134] A person skilled in the art can easily figure out other implementations of the present specification after considering and practicing the present specification disclosed here. The present specification is intended to cover any variations, usage, or adaptations of the present specification that follow the general principles of the present specification and include common general knowledge or commonly used technical means in the art that are not disclosed in the present specification. The present specification and implementations are merely examples. The protection scope and spirit of the present specification are indicated by the following claims.

[0135] It should be understood that the present specification is not limited to the precise structures described above and illustrated in the accompanying drawings, and various modifications and changes can be made without departing from the scope thereof. The protection scope of the present specification should be defined by the appended claims.

[0136] The above descriptions are merely preferred implementations of one or more implementations of the present specification, and are not intended to limit the present specification. Any modification, equivalent replacement, improvement, etc., made without departing from the spirit and principles of the present specification shall fall within the protection scope of the present specification.

CLAIMS

1. A blockchain-based data verification method, comprising:

obtaining (102) a data identifier of target data published by a data provider in
5 a blockchain, wherein the data identifier indicates a storage location of the target data
in the blockchain;

determining (104) whether the target data has been stored in the blockchain based
on the data identifier;

in response to determining that no target data corresponding to the data
10 identifier is stored in the blockchain, indicating that the data identifier published by the
data provider is a false data identifier;

in response to determining that the target data has been stored in the
blockchain, obtaining (106) the target data from the storage location indicated by
the data identifier, wherein the target data stored in the blockchain comprises
15 verification information used to verify authenticity of the target data and auxiliary
information related to authenticity verification, wherein the verification
information is a first hash value obtained by performing a hash calculation based
on original content of the target data and the auxiliary information, wherein the
auxiliary information comprises an identifier of a data uploader, an upload
20 timestamp of the target data, and an upload location of the target data, wherein
the upload location includes an address description;

executing an installed verification program to verify authenticity of the obtained
target data based on the verification information;

performing a hash calculation on the original content of the target data and the
25 auxiliary information to obtain a second hash value;

determining whether the second hash value is the same as the first hash value
carried in the target data stored in the blockchain;

if yes, determining that the obtained target data passed the authenticity
verification, or if not, determining that the obtained target data has failed to pass the
30 authenticity verification; and

outputting a verification result to a data verifier.

2. The method according to claim 1, wherein the target data published by the data
provider carries a graphic code generated based on the data identifier;

35 and

obtaining (102) a data identifier of target data published by a data provider in a blockchain comprises:

scanning and parsing the graphic code to obtain the data identifier of the target data in the blockchain.

5

3. The method according to claim 1 or 2, wherein the data identifier is a transaction hash value of a transaction of the target data; and
determining (104) whether the target data has been stored in the blockchain based on the data identifier comprises:

10 performing Simplified Payment Verification, SPV, for the transaction based on a transaction hash value of the transaction of the target data, to determine whether the transaction has been stored in the blockchain.

4. The method according to claim 1, further comprising:

15 outputting the target data obtained from the storage location indicated by the data identifier to the data verifier, so that the data verifier compares the outputted target data with the target data published by the data provider.

5. A blockchain-based data verification apparatus (40), comprising:

20 a first acquisition module (401), configured to obtain (102) a data identifier of target data published by a data provider in a blockchain, wherein the data identifier indicates a storage location of the target data in the blockchain;

a determining module (402), configured to determine (104) whether the target data has been stored in the blockchain based on the data identifier, and if it is
25 determined that no target data corresponding to the data identifier is stored in the blockchain, indicate that the data identifier published by the data provider is a false data identifier;

a second acquisition module (403), configured to: if the target data has been stored in the blockchain, obtain (106) the target data from the storage location indicated
30 by the data identifier, wherein the target data stored in the blockchain comprises verification information used to verify authenticity of the target data and auxiliary information related to authenticity verification,

wherein the verification information is a first hash value obtained by performing a hash calculation based on original content of the target data and the auxiliary
35 information, wherein the auxiliary information comprises an identifier of a data

uploader, an upload timestamp of the target data, and an upload location of the target data, wherein the upload location includes an address description; and a verification module (404), configured to:

execute an installed verification program to verify authenticity of the obtained target

5 data based on the verification information,

perform a hash calculation on the original content of the target data and the auxiliary information to obtain a second hash value,

determine whether the second hash value is the same as the first hash value carried in the target data stored in the blockchain,

10 if yes, determine that the obtained target data passed the authenticity verification, or if not, determine that the obtained target data has failed to pass the authenticity verification, and

output a verification result to a data verifier.

15 6. The apparatus (40) according to claim 5, wherein the target data published by the data provider carries a graphic code generated based on the data identifier; and

the first acquisition module (401) is configured to:

scan and parse the graphic code to obtain the data identifier of the target data in the
20 blockchain.

7. The apparatus (40) according to claim 5 or 6, the data identifier is a transaction hash value of a transaction of the target data; and

the determining module (402) is configured to:

25 perform Simplified Payment Verification, SPV, for the transaction based on a transaction hash value of the transaction of the target data, to determine whether the transaction has been stored in the blockchain.

30 8. The apparatus (40) according to claim 5, wherein the second acquisition module (403) is further configured to:

output the target data obtained from the storage location indicated by the data identifier to the data verifier, so that the data verifier compares the outputted target data with the target data published by the data provider.