(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
31 August 2017 (31.08.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/145048 Al

(51) International Patent Classification:
H04L 9/08 (2006.01) *H04L 9/30* (2006.01)

(21) International Application Number:
PCT/IB2017/050979

(22) International Filing Date:
21 February 2017 (21.02.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

1603117.1	23 February 2016 (23.02.2016)	GB
1604497.6	16 March 2016 (16.03.2016)	GB
1619301.3	15 November 2016 (15.11.2016)	GB

(71) Applicant: **NCHAIN HOLDINGS LIMITED** [AG/AG]; Fitzgerald House, 44 Church Street, St. John's (AG).

(72) Inventor: **WRIGHT, Craig Steven;** c/o Urquhart-Dykes & Lord LLP, 7th Floor Churchill House, Churchill Way, Cardiff CF10 2HH (GB).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available):* AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available):* ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

WO2017/145048 A1

(54) Title: CRYPTOGRAPHIC METHOD AND SYSTEM FOR SECURE EXTRACTION OF DATA FROM A BLOCKCHAIN

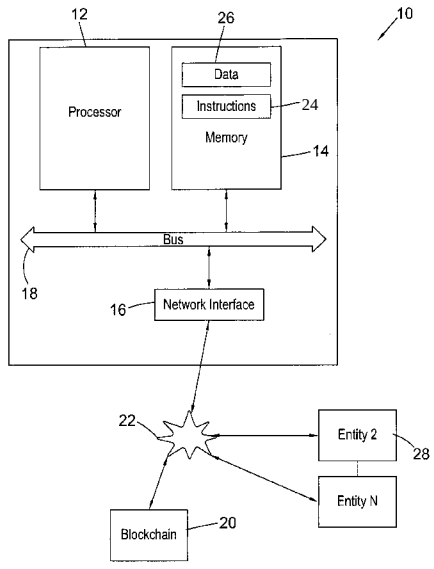

Fig. 1

(57) Abstract: The invention relates generally to cryptographic techniques for secure processing, transmission and exchange of data. It also relates to peer-to-peer distributed ledgers such as (but not limited to) the Bitcoin blockchain. In particular, it relates to control solutions for identifying, protecting, extracting, transmitting and updating data in a cryptographically controlled and secure manner. It also relates to system inter- operability and the ability to communicate data between different and distinct computing systems. The invention provides a computer implemented method (and corresponding systems) comprising the steps of identifying a set of first structure public keys comprising at least one public root key associated with a first structure of interest of an entity and one or more associated public sub-keys; deriving a deterministic association between the at least one public root key and the one or more associated public sub-keys; and extracting data from a plurality of transactions (TXs) from a blockchain. The data comprises data indicative of a blockchain transaction (Tx) between the first structure and at least one further structure; and a first structure public key associated with the first structure. The first structure public key is part of a cryptographic public/private key. The method includes the step of generating an output for the first structure of interest by matching at least part of the set of first structure public keys to the extracted transaction data using the deterministic association. The one or more public sub-keys is generated or determined using Elliptic Curve Cryptography (ECC) and a deterministic key (DK) that is based on a cryptographic hash of a message (M). The one or more public sub-keys is determined based on a scalar addition of an associated public parent key and the scalar multiplication of a deterministic key (DK) and a generator (G).

Cryptographic Method and System for Secure Extraction of Data From a Blockchain

Technical Field

[0001] The present disclosure relates generally to cryptographic techniques for secure processing, transmission and exchange of data. It also relates to peer-to-peer distributed ledgers such as (but not limited to) the Bitcoin blockchain and associated technologies. In particular, it relates to control solutions for identifying, protecting, extracting, transmitting and updating data in a cryptographically controlled and secure manner. It also relates to system inter-operability and the ability to communicate data between different and distinct computing systems.

Background

[0002] A blockchain is a peer-to-peer, electronic ledger which is implemented as a computer-based decentralised, distributed system made up of blocks which in turn are made up of transactions. Each transaction is a data structure that encodes the transfer of control of a digital asset between participants in the blockchain system, and includes at least one input and at least one output. Each block contains a hash of the previous block to that blocks become chained together to create a permanent, unalterable record of all transactions which have been written to the blockchain since its inception. Transactions contain small programs known as scripts embedded into their inputs and outputs, which specify how and by whom the outputs of the transactions can be accessed. On the Bitcoin platform, these scripts are written using a stack-based scripting language.

[0003] In order for a transaction to be written to the blockchain, it must be "validated". Network nodes (miners) perform work to ensure that each transaction is valid, with invalid transactions rejected from the network. Software clients installed on the nodes perform this validation work on an unspent transaction (UTXO) by executing

its locking and unlocking scripts. If execution of the locking and unlocking scripts evaluate to TRUE, the transaction is valid and the transaction is written to the blockchain. Thus, in order for a transaction to be written to the blockchain, it must be i) validated by the first node that receives the transaction - if the transaction is validated, the node relays it to the other nodes in the network; and ii) added to a new block built by a miner; and iii) mined, i.e. added to the public ledger of past transactions.

Although blockchain technology is most widely known for the use of cryptocurrency implementation, digital entrepreneurs have begun exploring the use of both the cryptographic security system Bitcoin is based on and the data that can be stored on the Blockchain to implement new systems. It would be highly advantageous if the blockchain could be used for automated tasks and processes which are not limited to the realm of cryptocurrency. Such solutions would be able to harness the benefits of the blockchain (e.g. a permanent, tamper proof records of events, distributed processing etc) while being more versatile in their applications.

One area of current research is the use of the blockchain for the implementation of "smart contracts". These are computer programs designed to automate the execution of the terms of a machine-readable contract or agreement. Unlike a traditional contract which would be written in natural language, a smart contract is a machine executable program which comprises rules that can process inputs in order to produce results, which can then cause actions to be performed dependent upon those results. Another area of blockchain-related interest is the use of 'tokens' (or 'coloured coins') to represent and transfer real-world entities via the blockchain. A potentially sensitive or secret item can be represented by the token which has no discemable meaning or value. The token thus serves as an identifier that allows the real-world item to be referenced from the blockchain.

[0004] In the present disclosure, these techniques are used in a novel and advantageous arrangement which allows data and assets to be transferred between different computer systems, one of which is a blockchain. The computer systems used within large organisations or entities can often be extremely large, complex and technically diverse. Security of the data is typically paramount. Access to, and utilisation of, the data stored in the systems needs to be efficient and cost effective. There is a need to overcome at least some of the technical difficulties which arise from the use of complex computing systems within large organisations, which are often hierarchical in nature.

[0005] In this document we use a financial accounting context to illustrate one possible use or application of the invention. However, it is important to note that this example is for illustrative purposes only and the invention is not limited in this regard. Instead, the invention provides a generalised, cryptographically enforced, blockchain-based solution for use with any type of non-blockchain implemented computing system associated with an organisation.

[0006] In relation to our illustrative use, a general ledger of an entity is a set of accounts that summarises all financial transactions occurring within an entity. The structure of this set of accounts is typically set up by the individual entity itself, and may include accounts reported on a business financial statement, such as cash, accounts payable, expense accounts as well as purchases and loans. In some situations, a general ledger may be subdivided into multiple sub-ledgers. In such a case, each sub-ledger within the general ledger may maintain a separate balance or a running balance of the financial position based on a monetary exchange for each account which the entity needs to track.

[0007] The accounting setup for entities can get relatively complex with hundreds or thousands of separate reporting points, i.e. business units, departments, products etc. For example, parallel books may be required: one by product line and one by department.

[0008] Large entities may implement complex financial systems based on large database structures to manage the accounts for the general ledger. For example, a financial system may require the management and synchronisation of various databases so that the captured data can be accurately reported.

[0009] In this document we use the term 'blockchain' to include all forms of electronic, computer-based, distributed ledgers. These include, but are not limited to, peer-to-peer, consensus-based blockchain and transaction-chain technologies, permissioned and un-permissioned ledgers, shared ledgers and variations thereof. The most widely known application of blockchain technology is the Bitcoin ledger, although other blockchain implementations have been proposed and developed. While Bitcoin may be referred to herein for the purpose of convenience and illustration, it should be noted that the invention is not limited to use with the Bitcoin blockchain and alternative blockchain implementations and protocols fall within the scope of the present invention.

Summary

[0010] The present invention is defined in the appended claims.

[001 1] The invention may provide a cryptographic method and corresponding system. It may provide a blockchain-implemented method/system. It may provide a control system for the secure identification, extraction, transmission, processing and/or update of data. It may provide a method/system of using cryptographic keys to integrate a blockchain with a non-blockchain implemented computing (e.g. data storage/processing) resource. It may provide a method/system of using cryptographic keys to extract data from a blockchain and/or integrate that blockchain- sourced data into a non-blockchain implemented storage resource. The invention may provide a computer implemented method of generating records for a (first) structure of interest of an entity on or from a peer-to-peer distributed ledger (i.e. blockchain). The entity may be referred to as an organisation, system or network.

[0012] The invention provides significant advantages over the prior art, not least because it allows the secure handling of data, the integrity of which is preserved through the use of novel cryptographic techniques. It also enables communication and exchange of data between separate computer systems which have been implemented upon different designs, structures, architectures and protocols. Thus, it provides system interoperability. It allows the use of cryptographic keys to serve as a facilitator or interface between an existing computer system and a blockchain, without the need for any adaptation of the protocols or structure of either. It provides enhanced performance and efficiency of data processing because it facilitates the management and synchronisation of potentially numerous and complex systems e.g. databases.

[0013] An entity may be made up of a plurality of elements e.g. system components, accounts, data records or network entities, etc. The entity may be a hierarchical entity having comprising a plurality of elements organised or associated in a hierarchical relationship. The elements within an entity may form a structure of elements. It may be a hierarchical structure. The structure may comprise a chain or tree-like hierarchy defining or being reflective of the relationships or associations between the elements in the entity. The elements in a structure may be associated with a sub-entity, unit or other association within the entity.

[0014] The entity and/or one or more elements within the structure may be associated with a respective cryptographic key. The key may form part of a public/private key pair. There may be one key or key pair which is designated as the "root" key/pair. Further elements in the structure may be associated with sub-keys or pairs which are derived from the root. The sub-keys may be generated in a deterministic manner. The sub-key may be generated or determined substantially as described within the section below entitled "method of sub-key generation". A sub-key may be generated, derived or determined based on another (preceding) key. Generation of the sub-key may comprise the use of ECC techniques. A sub-key may be generated, derived or determined using a deterministic key (DK) that is based on a cryptographic hash of a message (M). The message may be random, pseudo-random, pre-defined or selected. In a preferred embodiment, the message is selected, arranged or created to correspond

to a meaningful value such as, for example, an account number. The message may have some meaning in relation to the entity or a sub-entity/element. The message may provide a link, association or reference to the entity or element. A sub-key may be determined based on a scalar addition of the associated public parent key and the scalar multiplication of the deterministic key and a generator (G). The message may be stored within or as metadata in a blockchain transaction (Tx). The message may be rehashed in order to provide a further sub-key.

[0015] The invention may comprise the steps of:
associating one or more elements of a structure with a cryptographic sub-key derived from another cryptographic key;

[0016] Extracting, from a blockchain transaction (Tx), data comprising or relating to a sub-key of an element; and
transmitting the extracted data to a non-blockchain based computer system. The non-blockchain based computer system may not be part of the blockchain.

[0017] The method may comprise the step of scanning, traversing and/or analysing the blockchain to identify one or more blockchain transactions (TXs) which comprise the data relating to or comprising a cryptographic sub-key of an element.

The method may further comprise the steps of processing the extracted data to generate an output. In one embodiment, the output may be a result or a computation, or a report, or a decision, or a record such as a financial accounting record. It may comprise the step of communicating the output to a destination over a network.

[0018] Additionally or alternatively, the invention may comprise the steps:

identifying a set of first structure public keys comprising at least one public root key associated with the first structure and one or more associated public sub-keys;

deriving a deterministic association between the at least one public root key and the one or more associated public sub-keys;

extracting (copying) transaction data/information from a plurality of blockchain transactions from a peer-to-peer (P2P) distributed ledger (blockchain), the extracted data comprising at least:

data indicative of a transaction between the first structure and at least one further structure; and

a first structure public key associated with the first structure wherein the first structure public key is part of a cryptography pair comprising the first structure public key and a first structure private key; and

generating an output (e.g. financial accounting record) for the first structure of interest by matching the set of first structure public keys to the extracted transaction data using the deterministic association.

[0019] The transaction between the first structure and at least one further structure may involve the transfer an asset from one party to another. The asset may be a tokenised asset, or some sort of digital asset. It may be a portion of cryptocurrency. The transaction may transfer ownership and/or control of the asset from one party to another.

[0020] The blockchain may be associated with a cryptocurrency. For example, the blockchain may be the Bitcoin block chain. Alternatively, another cryptocurrency or block chain protocol may be used.

[0021] The first structure may represent a defined group of items, e.g. components within the entity. The items could be physical or logical e.g. accounts. For example, the first structure may be associated with a first business unit, such as a first group of

users, a department or a team; a first product or service of the entity; or the complete entity.

[0022] The at least one further structure may be within the entity. For example, the at least one further structure may be associated with a second business unit, or a second product or service.

[0023] Alternatively, the at least one further structure may be within a further entity. For example, the at least one further structure may be associated with a business unit of a further entity, a product or service of the further entity, or the complete further entity.

[0024] The set of first structure public keys may comprise the at least one public root key and all associated public sub-keys.

[0025] The step of deriving the deterministic association between the at least one public root key and the one or more associated public sub-keys may comprise determining a rule for determining the public sub-keys from an associated parent key. For example, the rule to determine a sub-key may comprise determining a hash of a message to create a seed value.

[0026] The deterministic association between the at least one public root key and the one or more associated public sub-keys may be a tree hierarchy. Alternatively, the deterministic association may be a chain hierarchy.

[0027] Each public sub-key may represent a virtual or physical or logical item (e.g. an account) within the entity.

[0028] The step of identifying the set of first structure public keys may comprise identifying the at least one root key and determining the one or more sub-keys based on the at least one root key. The at least one root key may be identified by obtaining data from an internal database, such as for example a chart of accounts.

[0029] The step of extracting the transaction data from the plurality of transactions from the blockchain may further comprise extracting or copying one or more of the following data items from the plurality of transactions:

a transaction input value;

a transaction output value;

a rule for deriving the transaction input or output value based on the data indicative of the transaction;

a time stamp for the transaction; and

a block number of a block of the blockchain.

[0030] The transaction between the first structure and a further structure may relate to a currency transaction, an exchange of a contract, a goods or services transaction or the like. With regard to the currency transaction, it will be appreciated that the transaction may relate to a cryptocurrency transaction such as Bitcoin, or a fiat currency transaction for example by using a token amount of cryptocurrency.

[0031] The step of extracting the transaction information from the plurality of transaction records from the P2P distributed ledger may comprise identifying transaction records that have not been previously extracted to generate financial accounting records. In one example, the method may comprise a step of identifying a block number of a block of the P2P distributed ledger and comparing the block number with financial accounting records generated previously.

[0032] The method may comprise a step of posting the output to a computer-based resource. This may be any type of computer-based system, such as a database system, or financial accounting ledger for example. It may be internal to the entity in that it may be arranged and configured for storage of data relating to the entity and/or its sub-entities. . The computer-based resource may be a general ledger of the entity.

Alternatively, the financial accounting ledger may be a sub-ledger of a general ledger of the entity. As mentioned above, a general ledger of an entity typically represents a set of accounts that summarises all financial transactions occurring within the entity.

[0033] The step of posting the generated output to the computer-based resource may be conducted automatically. In other words, it may be an automated, computer-executed process, without the need for manual intervention.

[0034] The step of posting the generated output may be conducted on a periodic basis. This may be after a pre-determined period of time or at a prescribed time.

[0035] The step of posting the generated output may comprise writing the generated output to one or more locations or files. These may be referred to as, for example, posting files The one or more posting files may have any suitable format, such as JSON, XML or XBRL. The one or more posting files may be encrypted. For example, the one or more posting files may be hashed. In a further example, the one or more posting files may be encrypted using a calculated secret. An example for calculating a secret is described in further detail in the section below entitled "Method for Sub-key Generation".

[0036] The method may comprise a step of signing the one or more posting files using a first structure private key, wherein the first structure private key is part of an asymmetric cryptography pair comprising the first structure private key and an associated first structure public key. In this way, the one or more posting files can be validated using the first structure public key that is associated with the first structure private key used to sign the one or more posting files.

[0037] The method may comprise a step of storing the generated output. In one particular example, the generated output may be recorded on an internal database or other storage facility of or within the entity. In an alternative example, the generated output may be stored on a public database that may be centralised or distributed, such as a distributed hash table. In a specific example, the generated output may be stored

on blockchain in the form of a transaction (Tx). For example, the generated output may be recorded as metadata of (within) a blockchain transaction (Tx). A hash value of the transaction may be signed using a private key of the first structure of interest.

[0038] The method may comprise a step of encrypting the generated output for storing the data at a database or blockchain.

[0039] The method may comprise a step of hashing and/or signing the generated output for storing the generated output on a database, storage facility or blockchain.

[0040] The method may comprise a step of analysing the generated output. In one example, the method may comprise a step of determining a cash flow of the first structure of interest. In a further example, the method may comprise a step of determining assets and/or debts for the first structure of interest.

[0041] The step of posting the generated financial accounting records may comprise making the generated financial accounting records available to a dedicated accounting system. The accounting system may be configured to execute dedicated accounting software for the entity.

[0042] In accordance with embodiments of the present disclosure, there is provided a computer system arranged to implement any embodiment of a method of the invention. It may be arranged for generating records for a first structure of interest of an entity on a blockchain comprising:

a processing device configured to

identify a set of first structure public keys comprising at least one public root key associated with the first structure and one or more associated public sub-keys; and

derive a deterministic association between the at least one public root key and the one or more associated public sub-keys;

a network interface configured to extract transaction information from a plurality of transaction records from a peer-to-peer (P2P) distributed ledger, the extracted information at least:

information indicative of a transaction between the first structure and at least one further structure; and

a first structure public key associated with the first structure wherein the first structure public key is part of a cryptography pair comprising the first structure public key and a first structure private key; and

wherein the processing device is further configured to generate financial accounting records for the first structure of interest by matching the set of first structure public keys to the extracted transaction information using the deterministic association.

[0043] A computer program comprising machine-readable instructions to cause a processing device to implement any one of the method described above.

Brief Description of Drawings

[0044] Embodiments of the present disclosure will now be described, by non-limiting example only, with reference to the accompanying drawings, in which:

[0045] Fig. 1 is a schematic diagram of an example system to generate financial accounting records;

[0046] Fig. 2 is an alternative schematic diagram of an illustrative computer system to generate financial accounting records;

[0047] Fig. 3 is a flow chart of an illustrative computer-implemented method for determining accounting configuration to generate financial accounting records;

[0048] Fig. 4 is a schematic diagram illustrating a deterministic association between a root public key and one or more public sub-keys;

[0049] Fig. 5 is a flow chart of an illustrative computer-implemented method for extracting information from a P2P distributed ledger (blockchain) to generate financial accounting records; and

[0050] Fig. 6 is a flow chart of an illustrative computer-implemented method for posting the generated financial accounting records to a financial accounting ledger.

[0051] Figures 7 to 13 show various aspects of an illustrative technique for deriving sub-keys from a parent key as discussed below, this technique being suitable for use in relation to aspects of the present invention.

Description of Embodiments

Overview

[0052] The present disclosure generally relates to a method and a system which uses a peer-to-peer (P2P) distributed ledger (hereafter "blockchain"), such as the Bitcoin Blockchain, in the identification, extraction, transmission, processing and/or update of data. The invention provides a secure, cryptographically enforced solution for using blockchain-posted data to augment, add to or integrate with data in a computer-based resource provided within an organisation (entity). The computer-based resource may be a data base, an accounting system or any other type of computer-based facility for the storage and/or processing of data. This system may be referred to as an "internal" system as it is internal to the organisation rather than part of the blockchain.

The invention provides a data control and security solution by assigning cryptographic keys to elements associated with an entity or sub-entity. The elements are organised into a (linear or tree) structure which is defined or specified by the relationships

between the elements. Each element within the structure is associated with a cryptographic public/private key pair, with a root key pair at the top of the structure. Elements which are lower in the structure are associated with key pairs which are deterministically derived from a previous key-pair. This deterministic derivation may, preferably, be performed in accordance with a method substantially as described in the section below entitled "Method of sub-key generation". Thus a hierarchy of cryptographic key-pairs is generated to reflect the hierarchical relationship between elements within the entity. In the following illustration, the term "accounts" may be used instead of "element".

[0053] Purely for the sake of illustration, and without any intention to limit the nature of the invention to this use or application area, an example is now provided in which the invention is used to generate financial accounting records for a structure of interest of an entity. It is important to note that this financially-oriented aspect is not essential to the invention, and that the invention can be used in conjunction with any type of internal computer-based system, and for the control, transmission and security of any type of data.

[0054] In our example, the structure of interest may for example be a business unit, such as a department or sub-entity within the entity, or may represent the complete entity for which the output (e.g. accounting records) are to be generated. Alternatively, the structure of interest may relate to a product or service of interest or any suitable structure of the entity for which financial accounting records are to be generated and optionally posted to, for example, the general ledger of the entity.

[0055] Embodiments of the present disclosure provide significant advantages. For example, it is an advantage to utilise a public blockchain, such as the Bitcoin Blockchain, to generate the record of financial transactions as the Bitcoin Blockchain is inherently decentralised. This means that the transaction records on the Bitcoin Blockchain are stored synchronously across the network which ensures that all information is distributed and publicised.

[0056] Furthermore, the generated financial accounting records for the structure of interest may be recorded on the same blockchain, for example in the form of metadata in a transaction record. This will provide the advantage that a permanent, immutable and public record is formed which is a true and accurate reflection of the activities of an entity. This may ultimately remove the requirement of the assistance of external parties, such as auditors.

[0057] Referring now to Figure 1, there is shown a computer system 10 for generating accounting records in accordance with illustrative embodiments of the present disclosure. The financial accounting records may for example be posted to a financial accounting ledger, such as the general ledger of the entity.

[0058] In the present example, the computer system includes a processor 12 arranged to control and coordinate operations, a memory 14, and a network interface 16 that communicate with each other via a bus 18. The network interface 16 facilitates wireless communication between the computer system 10 and a P2P distributed ledger, such as the Bitcoin Blockchain 20, through a communications network, which in this example is the Internet 22.

[0059] The memory 14 stores instructions 24 and data 26 for the processes as described in the following, and the processor 12 performs the instructions 24 from the memory 14 to implement the processes. It should be noted that although the computer system 10 is shown as an independent network element, the computer system 10 may alternatively be part of another network element and functions performed by the computer system 10 may be distributed between multiple network elements. The computer system 10 may represent a computer system of a first entity.

[0060] Figure 1 further shows a further computer system 28 of a second entity. This is for illustrative purposes only and it will be appreciated that other computer systems or entities may be part of the network.

Financial accounting ledger for an entity / general ledger

[0061] As mentioned above, a general ledger is a set of accounts that summarises all financial transactions occurring within an entity. It is typically used as the main financial record of an entity and incorporates all financial transactions (accounts) relating to an entity. These may include the asset accounts (for example, cash and cash equivalents, securities, accounts receivable, inventory and intellectual property) and liability accounts (for example, notes payable, accounts payable, expense accounts, loans). The above examples of accounts are merely exemplary and not to be taken as restrictive. Each account in the general ledger traditionally maintains a balance of the financial position of the entity based on monetary exchange for each account.

[0062] While both small and large entities need to maintain a record of the accounts by way of the general ledger, larger entities may need to track thousands or even hundreds of thousands of accounts. Such tracking may become a great and potentially complex task. There are several difficulties that come with maintaining a very large general ledger with such ledgers possibly taking days to audit or balance.

[0063] Further to this, some organisations may consider introducing sub-ledgers within the general ledger to simplify the complexity of the general ledger.

[0064] General ledgers may be the subject to internal and external audits of an entity. Auditing is often a tedious and time-consuming task that is performed by specially designated auditors. However, by recording the general ledger or one or more of the sub-ledgers on a publicly viewable blockchain, this potentially facilitates and alleviates the tasks of auditors.

[0065] In the following, the term "financial accounting ledger" may refer to the general ledger of an entity or to one or more of the sub-ledgers within the general ledger. However, the invention is not limited to use only with financial or accounting systems, and any other type of computer-based system may be used.

The Bitcoin Blockchain

[0066] Whilst embodiments described below may refer specifically to transaction records on the Bitcoin Blockchain (or simply the Blockchain), it will be appreciated that the present disclosure may be implemented using any blockchain. The Bitcoin Blockchain is used below to describe aspects of the present disclosure for simplicity only due to its high level of standardisation and large quantity of associated public documentation.

[0067] The Blockchain is a public transaction ledger which is distributed across networked nodes participating in a system based on the Bitcoin protocol. Each Bitcoin transaction is broadcast to the network, the transactions are confirmed and then aggregated into blocks. These blocks are then included on the Blockchain.

[0068] A full copy of a blockchain contains every transaction (Tx) posted to the ledger since its inception. Thus, a continuously growing list of transaction records is provided. Since each transaction entered onto the Blockchain is cryptographically enforced, the Blockchain is hardened against tampering and revision, even by operators of the data store nodes.

[0069] Due to the transparency of the Bitcoin Blockchain, transaction histories are publicly available for each transaction. It is a further advantage of the Blockchain that the transaction and the record of the transaction are the same, i.e. the record of the transaction is embedded within the transaction.

[0070] In this way, the information relating to the data exchange is captured in the actual transaction(Tx). This record is permanent and immutable, and each transaction that is conducted using Bitcoin is therefore not only facilitated by the Blockchain, but also immutably recorded in the Blockchain. This therefore removes the requirement for a third party to keep the transaction record on a separate database.

[0071] In embodiments of the present disclosure, instead of or in addition to being used in its designed function of storing transaction records representing payments of Bitcoins from one party to another, the Blockchain is used in a novel manner to

generate data for a structure of interest of an entity or organisation, such as a business unit or a product of interest. The data can be used in any foreseeable manner such as, for example, the generation of accounting records. For the sake of simplicity, but without limitation, the following illustration will refer to "accounting records". In this example, the generated accounting records are suitable for posting to the general ledger of an entity, or feeding to a dedicated accounting system of the entity.

[0072] While the exemplary embodiments below refer to the Bitcoin Blockchain as the public ledger, it is to be understood the disclosure also applies to any blockchain platform or protocol.

Functional components of the computer system

[0073] A representation of an example implementation of a computer system 40 is shown in Figure 2, with functional components of the computer system 40 shown instead of hardware components. The functional components in this example may be implemented using the hardware components of the computer system 10 shown in Figure 1 such that network interfaces are provided for facilitating access to the Bitcoin Blockchain.

[0074] In this example, the computer system 40 comprises a control unit 42 for controlling and coordinating operations of the components of the computer system 40. This control unit 42 may for example be implemented by the processor 12 shown in Figure 1.

[0075] Further, the computer system 40 has a network interface 44 for facilitating access to information stored on the Bitcoin Blockchain 20 through the Internet 22.

[0076] The computer system 40 further comprises a data base management system ("DBMS") 50 that stores data that is received and/or generated in the computer system 40 in a data storage 52. It will be understood that the data may alternatively be stored

in a remote database, such as in a cloud storage and can be received at the computer system 40 through the Internet 22 via the network interface 44.

[0077] The data storage 52 comprises a memory for storing accounting configuration data 54 of the first entity. For example, the accounting configuration data 54 may comprise a chart of accounts which collects account names and identifiers for each account. This may further include information on the public root keys as will be described in further detail with reference to Figure 4.

[0078] The stored accounting configuration data 54 may be accessed by an account configuration determiner 56 of the computer system 40.

[0079] The data storage 52 further comprises a memory for storing posting files 58 that are created by a posting component 60 of the computer system 40. For example, generated financial accounting records may be written to one or more posting files that can be stored in the data storage 52.

[0080] The computer system 40 further comprises a transaction information collector 62 that is in communication with the network interface 44 to obtain transaction information from transaction records on the Bitcoin Blockchain 20 via the Internet 22.

[0081] A matching component 64 of the computer system 40 is in communication with the account configuration determiner 56 and the transaction information collector 62. The matching component 64 is configured to match extracted transaction information from the Blockchain to identified public keys associated with respective accounts of the entity to generate financial accounting records.

[0082] As mentioned above, these generated financial accounting records may then be written to one or more posting files and can be posted to the general ledger 66 of the entity in the data storage 52.

Overview of illustrative method to generate financial accounting records

[0083] In the following, the method of generating financial accounting records is divided into four main method steps, i.e. determining 100 accounting configuration, extracting 200 transaction information from transaction records on the Blockchain, generating the financial accounting records and creating 300 one or more posting files, and posting 400 the one or more posting files to the general ledger of the entity.

[0084] The first structure of interest may be any suitable structure of the first entity for which accounting records are to be generated. For example, the first structure may relate to the first entity in its entirety and generated accounting records will be posted to the general ledger of the first entity. In another example, the first structure of interest may represent a business unit, such as a department, a team or a specified group of employees. Alternatively, the first structure of interest may relate to a first product or service which may span across a plurality of business units. In this way, the generated accounting records can be created to analyse the accounts associated with a desired product or service.

Account configuration

[0085] Referring now to Figures 3 and 4, there is shown a flow chart illustrating method steps for determining 100 accounting configuration, and a schematic illustration of some of the method steps of method 100.

[0086] In particular, the method 100 may include a step of identifying a set of first entity public keys comprising at least one public root key associated with the first entity and one or more associated public sub-keys. Each of the sub-keys may represent an account within the first entity. In this regard, the set of first entity public keys may initially be identified by identifying 100.10 one or more accounts for which financial accounting records are to be generated.

[0087] In a further step, at least one public root key associated with the identified one or more accounts is identified 100.20. This may be done by accessing data stored on an internal data storage of the first entity, such as data storage 54 shown in Figure 2. The

internal data storage may store a chart of accounts which collects account names and identifiers for each account. The identifiers may include the associated public root key. In the example as shown in Figure 3, the method 100 identifies three public root keys which may represent accounts of three distinct business units.

[0088] The method 100 further comprises a step of determining 100.30 a deterministic association between the identified public root keys and one or more public sub-keys. In other words, a rule is determined to derive one or more public sub-keys that are dependent on the public root keys. In this particular example as shown in Figure 4, the deterministic association defines a tree hierarchy where each of the three identified public root keys is associated with successive child and grandchild tree nodes.

[0089] Following the determination of the deterministic association in step 100.30, one or more public sub-keys associated with the at least one root key are determined and some are selected using the information from the chart of accounts in step 100.10. It will be appreciated that depending on the identification of the accounts for which the accounting records are to be generated, all or only part of the sub-keys that are dependent on the public root key are determined.

[0090] In this particular example as shown in Figure 4, four public sub-keys are selected for the generation of financial accounting records; a first child of the first public root key, a first child and a first grandchild of the second public root key, and a second child or the third root key. The four public sub-keys may for example be associated with a specific product or service that spans across the three distinct business units represented by the root keys.

[0091] In one specific example, a series of successive deterministic public keys may be determined where each successive public key may be determined based on the preceding public key. For example, Elliptic Curve Cryptography (ECC) may be used. In this regard, a sub-key may be determined by initially determining a deterministic key (DK) that is based on a cryptographic hash of a message (M) which may be random,

pseudo random, or pre-defined. A sub-key may then be determined based on a scalar addition of the associated public parent key and the scalar multiplication of the deterministic key (DK) and a generator (G). In order to re-determine the deterministic key (DK) and thereby the sub-keys, only the message (M) and the generator (G) are necessary. In this regard, the message (M) may be stored in the form of metadata of a transaction record on the Blockchain, for example in the form of an invoice number.

[0092] In a further example, the message may be rehashed for determining the next generation public sub-key.

[0093] The steps for determining successive public keys as briefly described above are described and illustrated in further detail below in the section entitled "method of sub-key generation".

[0094] In a further step 100.40, one or more rules are determined for obtaining information to be included in the financial accounting records for the selected public keys. In other words, method step 100.40 determines what information will be required for the accounting records to be generated for the general ledger and how this information will be obtained.

[0095] In particular, the information may include:

Transaction input / output;

The General Ledger account to post to

The General Ledger account to counter-post to

A rule for calculating the value for the transaction

A rule for deriving the narrative for the transaction

Extracting information from the Blockchain

[0096] Referring now to Figure 5, there is shown a flow chart illustrating a method 200 including method steps for extracting 200 information from the Bitcoin Blockchain.

[0097] Initially, the method 200 comprises a step of identifying 200.10 one or more blocks of the Blockchain for extracting transaction information from a plurality of transactions that are recorded on the Blockchain. For example, the step of identifying 200.10 the one or more blocks may comprise identifying a first block number associated with a block that contains transaction records that have not previously been used for generating financial accounting records. This may be done by identifying the relevant block number of the block which is part of each transaction record on the Blockchain. Alternatively, the one or more blocks may be identified based on a transaction date or time. As described above, the blocks on the Bitcoin Blockchain have a defined order and each block includes a hash of the previous block.

[0098] Once the one or more blocks have been identified, a plurality of transaction records associated with the first structure of the entity is scanned 200.20 on the Blockchain to match against the sub-keys identified in step 100.30.

[0099] In a further step 200.30, the rules determined in step 100.40 are run to extract transaction information from each of the matched transaction records on the Blockchain. Thus, it will be determined what information needs to be extracted, how to derive certain information from the extracted information and where/how to post the information to the general ledger.

[0100] For example, one or more of the following information from the plurality of transaction records are extracted:

a transaction input value;

a transaction output value;

public keys;

metadata;

a rule for deriving the transaction input or output value based on the information indicative of the transaction;

a time stamp for the transaction; and

a block number of a block of the P2P distributed ledger.

[0101] Once the transaction records within one block have been scanned and processed, the next block in the Blockchain will be scanned for transaction records associated with the first entity.

[0102] The extracted information from step 200.30 is then used to generate 200.40 the financial accounting records.

Transaction records

[0103] In the following, a transaction record stored on the Bitcoin Blockchain will be considered in further detail.

[0104] Each transaction record on the Bitcoin Blockchain comprises at least a first structure public key associated with the first structure. This identifies that the first structure of the entity is involved in the transaction that is stored on the Bitcoin Blockchain.

[0105] Further, each transaction (Tx) on the Bitcoin Blockchain comprises at least information indicative of a transaction between the first entity and a further entity. It should be noted that any type of entity may be exchangeable between the first entity and the further entity and stored in a transaction.

[0106] Examples of transaction may include cryptocurrency exchanges, such as Bitcoin transactions, fiat transactions, tokens (representing any type of transferrable contract or asset), contracts, and goods and services of any type.

Tokens

[0107] Tokens may represent an asset, or a contract conferring specified rights upon the holder to be redeemed for fiat currency (virtual banknotes), to denote ownership of property (e.g., title deeds), or grant access to an event (tickets), to name a few of many examples. Goods and services may include new or second hand products, labour (e.g., charged by the hour), complete jobs (e.g., mowing the lawn), to name a few of many examples.

[0108] A token is an exchangeable entity represented by/representative of a contract. The contract may take one of several forms. For example, the contract may confer a right upon the holder or denote ownership of property. The value of the token may be contractually specified and is linked to the underlying BTC amount via a 'pegging rate' . The token is exchangeable via novel type of transaction using a cryptocurrency protocol such as the bitcoin protocol. The bitcoin value on the transaction acts as a token representing a rights contract in digital form. The contract itself may be stored on the transaction or may be kept in a publicly accessible location, or may be held privately by the parties to the contract depending on the particular embodiment. Where the contract is not stored on the transaction, the transaction may store a unique pointer to the contract.

[0109] Tokens may be divisible. A divisible token is one in which the value on the transaction output can be subdivided into smaller amounts which can be allocated across multiple new tokens. Examples of divisible tokens include tokens for fiat currency or for shares in a race horse. Divisible contracts may be defined as those that specify a non-zero pegging rate. In other words, the token value is tied to the underlying bitcoin value. Alternatively, tokens may be non-divisible. A non-divisible token is a contract that specifies the holder's rights in terms of a fixed value, e.g. a

contract to redeem a house or AU$1000. Non-divisible tokens are therefore not linked to the value of the underlying bitcoin.

[01 10] Tokens must be digitally signed by a token issuer to be valid. The issuer may, for example be an authority such as a Registrar of Title deeds. The issuer may issue a token to a user in return for payment. That token may then give the user the right to exercise the contract linked to the token, whether the contract represents the right to redeem fiat currency or for a service to performed.

[0111] Examples of tokens in accordance with the above, include the following:

- A fiat currency token that is pegged to the BTC value of the transaction output by the issuer of the contract. For example "the spender of this token (bitcoin transaction) is entitled to redeem any fraction of this token for Canadian dollars (CAD) at a rate of 1 share (lOcents) for every 1000 satoshis".
- A race horse owned by several members of a syndicate
- Any item where the ownership is by a title deed, e.g., a house or other property could be treated in this way.
- An electronic contract representing a concert ticket. This is inherently non-divisible.
- A bearer bond (non-divisible)
- A unique identifier attached to a Goods/service (such as a bar code or RFID). If used, this identifier is still preferably validated by a signature of an authorised entity; without a signature it will fall into the less secure 'goods/service' category (described below).
- A contract for the right to a service to be performed. It is noted that this is not the same as the actual service itself, but only the right to have the service performed for them. This right can be traded. For example, a voucher from Michael's Mowing for up to 3 hours lawn mowing within the Sydney metropolitan area. The holder of this voucher (contract) can redeem it for the actual service.

[01 12] Tokens must specify the value of a share, e.g., 1 share = 10 cents CAD, 1 share = 1 rupiah, or 1 share = 1% ownership of an item or property (race horse, house, etc.).

Posting files

[01 13] Referring now to Figure 6, there is shown a flow chart illustrating method steps of a method 300 of creating one or more posting files.

[01 14] The method 300 comprises a step of writing 300. 10 the generated accounting records from step 200.40 into one or more posting files. This may including consolidating specified information for posting purposes to the general ledger. For example, the generated accounting records may be consolidated by date, or each accounting record may represent an individual file.

[0115] The one or more posting files may have any suitable format, such as but not limited to JSON, XML or XBRL. The format may be predefined by the general ledger.

[0116] The method 300 may further comprise hashing 300.20 the one or more posting files. For example, a cryptographic hash algorithm may include SHA-256 to create a 256-bit posting file hash.

[01 17] The method 300 may further comprise a step of signing 300.30 the hash of the posting file. The hash of the posting file may, for example, be signed with a private key that is part of an asymmetric cryptography pair having the private key and an associated public key. In this way, the signed hash of the posting file can be validated using the associated public key.

[0118] Following the step of hashing 300.20 the one or more posting files, the hash is stored 300.40 on the Blockchain for permanent and unchangeable proof. For example, the hash of the posting file may be stored in the form of metadata of a transaction

record. As described above with reference to the P2SH, the posting file may be stored in the form of a "false" public key within the script.

[0119] In a specific example, the agent application may be used to record the hash of the posting file and the block number range on the Blockchain as the contract transaction.

Posting to General Ledger

[0120] In the following, method steps of method 400 are described of posting the one or more posting files to the general ledger of the first entity.

[0121] As described above, the general ledger is a set of accounts that summarises all transactions that occur within the entity. The method 400 of posting the one or more posting files to the general ledger may comprise accessing the one or more posting files in a secure manner and validating that the one or more posting files come from a recognised source. In this regard, the method 400 may comprise obtaining the public key that is associated with the private signing key used in step 300.30 to validate the signature.

[0122] Once the one or more posting files have been successfully validated, the one or more posting files are applied to the general ledger.

Dedicated accounting system

[0123] In one example, the financial accounting records are generated so that the accounting records can be fed into a dedicated accounting system of the entity. For example, the accounting system of the entity may be configured to execute a dedicated accounting software for the entity.

[0124] It will be appreciated that Bitcoin transactions may only form part of an entity's trading position. In such a case, the generated financial accounting records

utilising the transaction records on the Bitcoin Blockchain may be incorporated into an existing accounting model.

Method of Sub-key Generation

[0125] In accordance with the present invention there is a need to generate a sub-key from an original (master) key. A method for achieving this is now provided for illustration of one way in which this could be performed. The following provides an illustrative use of a technique which can be used in accordance with, or adapted for use with, the present invention. The following description is provided with reference to Figures 17 to 13.

Figure 7 illustrates a system 1 that includes a first node 3 which is in communication with a second node 7 over a communications network 5. The first node 3 has an associated first processing device 23 and the second node 5 has an associated second processing device 27. The first and second nodes 3, 7 may include an electronic device, such as a computer, phone, tablet computer, mobile communication device, computer server etc. In one example, the first node 3 may be a client (user) device and the second node 7 may be a server.

[0126] The first node 3 is associated with a first asymmetric cryptography pair having a first node master private key (Vic) and a first node master public key (Pic). The second node (7) is associated with a second asymmetric cryptography pair having a second node master private key (Vis) and a second node master public key (Pis). In other words, the first and second nodes are each in possession of respective public-private key pairs.

[0127] The first and second asymmetric cryptography pairs for the respective first and second nodes 3, 7 may be generated during a registration process, such as registration for a wallet. The public key for each node may be shared publicly, such as over communications network 5.

[0128] To determine a common secret (CS) at both the first node 3 and second node 7, the nodes 3, 7 perform steps of respective methods 300, 400 without communicating private keys over the communications network 5.

[0129] The method 300 performed by the first node 3 includes determining 330 a first node second private key (V_2c) based on at least the first node master private key (Vic) and a Generator Value (GV). The Generator Value may be based on a message (M) that is a shared between the first and second nodes, which may include sharing the message over the communications network 5 as described in further detail below. The method 300 also includes determining 370 a second node second public key (P_2s) based on at least the second node master public key (Pis) and the Generator Value (GV). The method 300 includes determining 380 the common secret (CS) based on the first node second private key (V_2c) and the second node second public key (P_2s).

[0130] Importantly, the same common secret (CS) can also be determined at the second node 7 by method 400. The method 400 includes determining 430 a first node second public key (P_2c) based on the first node master public key (Pic) and the Generator Value (GV). The method 400 further include determining 470 a second node second private key (V_2s) based on the second node master private key (Vis) and the Generator Value (GV). The method 400 includes determining 480 the common secret (CS) based on the second node second private key (V_2s) and the first node second public key (P_2c).

[0131] The communications network 5 may include a local area network, a wide area network, cellular networks, radio communication network, the internet, etc. These networks, where data may be transmitted via communications medium such as electrical wire, fibre optic, or wirelessly may be susceptible to eavesdropping, such as by an eavesdropper 11. The method 300, 400 may allow the first node 3 and second node 7 to both independently determine a common secret without transmitting the common secret over the communications network 5.

[0132] Thus one advantage is that the common secret (CS) may be determined securely and independently by each node without having to transmit a private key over a potentially unsecure communications network 5. In turn, the common secret may be used as a secret key (or as the basis of a secret key).

[0133] The methods 300, 400 may include additional steps. See Figure 8. The method 300 may include, at the first node 3, generating a signed message (SMI) based on the message (M) and the first node second private key (V_2c). The method 300 further includes sending 360 the first signed message (SMI), over the communications network, to the second node 7. In turn, the second node 7 may perform the steps of receiving 440 the first signed message (SMI). The method 400 also includes the step of validating 450 the first signed message (SM2) with the first node second public key (P_2c) and authenticating 460 the first node 3 based on the result of validating the first signed message (SMI). Advantageously, this allows the second node 7 to authenticate that the purported first node (where the first signed message was generated) is the first node 3. This is based on the assumption that only the first node 3 has access to the first node master private key (Vic) and therefore only the first node 3 can determine the first node second private key (V_2c) for generating the first signed message (SMI). It is to be appreciated that similarly, a second signed message (SM2) can be generated at the second node 7 and sent to the first node 3 such that the first node 3 can authenticate the second node 7, such as in a peer-to-peer scenario.

[0134] Sharing the message (M) between the first and second nodes may be achieved in a variety of ways. In one example, the message may be generated at the first node 3 which is then sent, over the communications network 5, the second node 7. Alternatively, the message may be generated at the second node 7 and then sent, over the communications network 5, to the second node 7. In some examples, the message (M) may be public and therefore may be transmitted over an unsecure network 5. One or more messages (M) may be stored in a data store 13, 17, 19. The skilled person will realise that sharing of the message can be achieved in a variety of ways.

[0135] Advantageously, a record to allow recreation of the common secret (CS) may be kept without the record by itself having to be stored privately or transmitted securely.

Method of registration 100, 200

[0136] An example of a method of registration 100, 200 will be described with reference to Figure 9, where method 100 is performed by the first node 3 and method 200 is performed by the second node 7. This includes establishing the first and second asymmetric cryptography pairs for the respective first and second nodes 3, 7.

[0137] The asymmetric cryptography pairs include associated private and public keys, such as those used in public-key encryption. In this example, the asymmetric cryptography pairs are generated using Elliptic Curve Cryptography (ECC) and properties of elliptic curve operations.

[0138] In the method 100, 200, this includes the first and second nodes agreeing 110, 210 on a common ECC system and using a base point (G). (Note: the base point could be referred to as a Common Generator, but the term 'base point' is used to avoid confusion with the Generator Value GV). In one example, the common ECC system may be based on secp256Kl which is an ECC system used by Bitcoin. The base point (G) may be selected, randomly generated, or assigned.

[0139] Turning now to the first node 3, the method 100 includes settling 110 on the common ECC system and base point (G). This may include receiving the common ECC system and base point from the second node 7, or a third node 9. Alternatively, a user interface 15 may be associated with the first node 3, whereby a user may selectively provide the common ECC system and/or base point (G). In yet another alternative one or both of the common ECC system and/or base point (G) may be randomly selected by the first node 3. The first node 3 may send, over the communications network 5, a notice indicative of using the common ECC system with a base point (G) to the second node 7. In turn, the second node 7 may settle 210 by

sending a notice indicative of an acknowledgment to using the common ECC system and base point (G).

[0140] The method 100 also includes the first node 3 generating 120 a first asymmetric cryptography pair that includes the first node master private key (Vic) and the first node master public key (Pic). This includes generating the first master private key (Vic) based, at least in part, on a random integer in an allowable range specified in the common ECC system. This also includes determining the first node master public key (Pic) based on elliptic curve point multiplication of the first node master private key (Pic) and the base point (G) according to the formula:

$$Pic = Vic \times G \qquad \text{(Equation 1)}$$

[0141] Thus the first asymmetric cryptography pair includes:

[0142] Vic : The first node master private key that is kept secret by the first node.

[0143] Pic: The first node master public key that is made publicly known.

[0144] The first node 3 may store the first node master private key (Vic) and the first node master public key (Pic) in a first data store 13 associated with the first node 3. For security, the first node master private key (Vic) may be stored in a secure portion of the first data store 13 to ensure the key remains private.

[0145] The method 100 further includes sending 130 the first node master public key (Pic), over the communications network 5, to the second node 7, as shown in figure 9. The second node 7 may, on receiving 220 the first node master public key (Pic), store

230 the first node master public key (Pic) in a second data store 17 associated with the second node 7.

[0146] Similar to the first node 3, the method 200 of the second 7 includes generating 240 a second asymmetric cryptography pair that includes the second node master private key (Vis) and the second node master public key (Pis). The second node master private key (Vis) is also a random integer within the allowable range. In turn, the second node master public key (Pis) is determined by the following formula:

[0147] Pis = Vis x G (Equation 2)

[0148] Thus the second asymmetric cryptography pair includes:

[0149] Vis : The second node master private key that is kept secret by the second node.

[0150] Pis: The second node master public key that is made publicly known.

[0151] The second node 7 may store the second asymmetric cryptography pair in the second data store 17. The method 200 further includes sending 250 the second node master public key (Pis) to the first node 3. In turn, the first node 3 may receive 140 and stores 150 the second node master public key (Pis).

[0152] It is to be appreciated that in some alternatives, the respective public master keys may be received and stored at a third data store 19 associated with the third node 9 (such as a trusted third party). This may include a third party that acts as a public directory, such as a certification authority. Thus in some examples, the first node master public key (Pic) may requested and received by the second node 7 only when determining the common secret (CS) is required (and vice versa).

[0153] The registration steps may only need to occur once as an initial setup.

[0154] Session initiation and determining the common secret by the first node 3

[0155] An example of determining a common secret (CS) will now be described with reference to Figure 10. The common secret (CS) may be used for a particular session, time, transaction, or other purpose between the first node 3 and the second node 7 and it may not be desirable, or secure, to use the same common secret (CS). Thus the common secret (CS) may be changed between different sessions, time, transactions, etc.

[0156] The following is provided for illustration of the secure transmission technique which has been described above.

Generating a message (M) 310

[0157] In this example, the method 300 performed by the first node 3 includes generating 310 a message (M). The message (M) may be random, pseudo random, or user defined. In one example, the message (M) is based on Unix time and a nonce (and arbitrary value). For example, the message (M) may be provided as:

$$\text{Message (M)} = \text{UnixTime} + \text{nonce} \qquad \text{(Equation 3)}$$

[0158] In some examples, the message (M) is arbitrary. However it is to be appreciated that the message (M) may have selective values (such as account number, Unix Time, etc) that may be useful in some applications.

[0159] The method 300 includes sending 315 the message (M), over the communications network 3, to the second node 7. The message (M) may be sent over an unsecure network as the message (M) does not include information on the private keys.

Determining a Generator Value (GV) 320

[0160] The method 300 further includes the step of determining 320 a Generator Value (GV) based on the message (M). In this example, this includes determining a cryptographic hash of the message. An example of a cryptographic hash algorithm includes SHA-256 to create a 256-bit Generator Value (GV). That is:

[0161] $GV = SHA\text{-}256(M)$ (Equation 4)

[0162] It is to be appreciated that other hash algorithms may be used. This may include other has algorithms in the Secure Hash Algorithm (SHA) family. Some particular examples include instances in the SHA-3 subset, including SHA3-224, SHA3-256, SHA3-384, SHA3-512, SHAKE 128, SHAKE256. Other hash algorithms may include those in the RACE Integrity Primitives Evaluation Message Digest (RIPEMD) family. A particular example may include RIPEMD- 160. Other hash functions may include families based on Zemor-Tillich hash function and knapsack-based hash functions.

Determining a first node second private key 330

[0163] The method 300 then includes the step 330 of determining 330 the first node second private key (V_2c) based on the second node master private key (Vic) and the Generator Value (GV). This can be based on a scalar addition of the first node master private key (Vic) and the Generator Value (GV) according to the following formula:

[0164] $V_{2C} = Vic + GV$ (Equation 5)

[0165] Thus the first node second private key (V_2c) is not a random value but is instead deterministically derived from the first node master private key. The corresponding public key in the cryptographic pair, namely the first node second public key (P_2c), has the following relationship:

$$P2C = V_2c \times G \quad \text{(Equation 6)}$$

[0166] Substitution of V_2c from Equation 5 into Equation 6 provides:

$$P_2c = (Vic + GV) \times G \quad \text{(Equation 7)}$$

where the ' + ' operator refers to elliptic curve point addition. Noting that elliptic curve cryptography algebra is distributive, Equation 7 may be expressed as:

$$P_2c = Vic \times G + GV \times G \quad \text{(Equation 8)}$$

[0167] Finally, Equation 1 may be substituted into Equation 7 to provide:

$$P_2C = Pic + GV \times G \quad \text{(Equation 9.1)}$$

$$Pic = Pic + SHA\text{-}256(M) \times G \quad \text{(Equation 9.2)}$$

[0168] Thus the corresponding first node second public key (Pic) can be derivable given knowledge of the first node master public key (Pic) and the message (M). The second node 7 may have such knowledge to independently determine the first node second public key (Pic) as will be discussed in further detail below with respect to the method 400.

Generate a first signed message (SMI) based on the message and the first node second private key 350

[0169] The method 300 further includes generating 350 a first signed message (SMI) based on the message (M) and the determined first node second private key (Vic). Generating a signed message includes applying a digital signature algorithm to digitally sign the message (M). In one example, this includes applying the first node second private key (Vic) to the message in an Elliptic Curve Digital Signature Algorithm (ECDSA) to obtain the first signed message (SMI). Examples of ECDSA include those based on ECC systems with secp256kl, secp256rl, secp384rl, se3cp521rl.

The first signed message (SMI) can be verified with the corresponding first node second public key (Pic) at the second node 7. This verification of the first signed message (SMI) may be used by the second node 7 to authenticate the first node 3, which will be discussed in the method 400 below.

Determine a second node second public key 370'

The first node 3 may then determine 370 a second node second public key (Pis). As discussed above, the second node second public key (Pis) may be based at least on the second node master public key (Pis) and the Generator Value (GV). In this example, since the public key is determined 370' as the private key with elliptic curve point

multiplication with the base point (G), the second node second public key (P_2s) can be expressed, in a fashion similar to Equation 6, as:

[0170] $P_{2S} = V_{2S} \times G$ (Equation 10.1)

[0171] $P_{2S} = Pis + GV \times G$ (Equation 10.2)

[0172] The mathematical proof for Equation 10.2 is the same as described above for deriving Equation 9.1 for the first node second public key (Pic). It is to be appreciated that the first node 3 can determine 370 the second node second public key independently of the second node 7.

Determine the common secret 380 at the first node 3

[0173] The first node 3 may then determine 380 the common secret (CS) based on the determined first node second private key (Vic) and the determined second node second public key (Pis). The common secret (CS) may be determined by the first node 3 by the following formula:

[0174] $S = Vic \times Pis$ (Equation 11)

[0175] Method 400 performed at the second node 7

[0176] The corresponding method 400 performed at the second node 7 will now be described. It is to be appreciated that some of these steps are similar to those discussed above that were performed by the first node 3.

[0177] The method 400 includes receiving 410 the message (M), over the communications network 5, from the first node 3. This may include the message (M) sent by the first node 3 at step 315. The second node 7 then determines 420 a Generator Value (GV) based on the message (M). The step of determining 420 the Generator Value (GV) by the second node 7 is similar to the step 320 performed by the

first node described above. In this example, the second node 7 performs this determining step 420 independent of the first node 3.

[0178] The next step includes determining 430 a first node second public key (P_2c) based on the first node master public key (Pic) and the Generator Value (GV). In this example, since the public key is determined 430' as the private key with elliptic curve point multiplication with the base point (G), the first node second public key (P_2c) can be expressed, in a fashion similar to Equation 9, as:

[0179] $P_2c = V_{2C} \times G$ (Equation 12.1)

[0180] $P_2c = Pic + GV \times G$ (Equation 12.2)

[0181] The mathematical proof for Equations 12.1 and 12.2 is the same as those discussed above for Equations 10.1 and 10.2.

[0182] The second node 7 authenticating the first node 3

[0183] The method 400 may include steps performed by the second node 7 to authenticate that the alleged first node 3, is the first node 3. As discussed previously, this includes receiving 440 the first signed message (SMI) from the first node 3. The second node 7 may then validate 450 the signature on the first signed message (SMI) with the first node second public key (P_2c) that was determined at step 430.

[0184] Verifying the digital signature may be done in accordance with an Elliptic Curve Digital Signature Algorithm (ECDSA) as discussed above. Importantly, the first signed message (SMI) that was signed with the first node second private key (V_2c) should only be correctly verified with the corresponding first node second public key (P2c), since V_2c and P_2c form a cryptographic pair. Since these keys are deterministic on the first node master private key (Vic) and the first node master public key (Pic) that were generated at registration of the first node 3, verifying first signed message (SMI) can be used as a basis of authenticating that an alleged first node sending the first

signed message (SMI) is the same first node 3 during registration. Thus the second node 7 may further perform the step of authenticating (460) the first node 3 based on the result of validating (450) the first signed message - see Figure 11.

[0185] The second node 7 determining the common secret

[0186] The method 400 may further include the second node 7 determining 470 a second node second private key (V_2s) based on the second node master private key (Vis) and the Generator Value (GV). Similar to step 330 performed by the first node 3, the second node second private key (V_2s) can be based on a scalar addition of the second node master private key (Vis) and the Generator Value (GV) according to the following formulas:

[0187] $V_{2S} = Vis + GV$ (Equation 13.1)

[0188] $V_{2S} = Vis + SHA\text{-}256(M)$ (Equation 13.2)

[0189] The second node 7 may then, independent of the first node 3, determine 480 the common secret (CS) based on the second node second private key (V_2s) and the first node second public key (P_2c) based on the following formula:

[0190] $S = V_2s \times P_2c$ (Equation 14)

[0191] Proof of the common secret (CS) determined by the first node 3 and second node 7

[0192] The common secret (CS) determined by the first node 3 is the same as the common secret (CS) determined at the second node 7. Mathematical proof that Equation 11 and Equation 14 provide the same common secret (CS) will now be described.

[0193] Turning to the common secret (CS) determined by the first node 3, Equation 10.1 can be substituted into Equation 11 as follows:

[0194] $S = V_{2C} \times P_{2}s$ (Equation 11)

[0195] $S = V_{2C} \times (V_{2S} \times G)$

[0196] $S = (V_{2C} \times V_{2S}) \times G$ (Equation 15)

[0197] Turning to the common secret (CS) determined by the second node 7, Equation 12.1 can be substituted into Equation 14 as follows:

[0198] $S = V_{2S} \times P_{2C}$ (Equation 14)

[0199] $S = V_{2S} \times (V_{2C} \times G)$

[0200] $S = (V_{2S} \times V_{2C}) \times G$ (Equation 16)

[0201] Since ECC algebra is commutative, Equation 15 and Equation 16 are equivalent, since:

[0202] $S = (V_{2C} \times V_{2S}) \times G = (V_{2S} \times V_{2C}) \times G$ (Equation 17)

[0203] The common secret (CS) and secret key

[0204] The common secret (CS) may now be used as a secret key, or as the basis of a secret key in a symmetric-key algorithm for secure communication between the first node 3 and second node 7.

[0205] The common secret (CS) may be in the form of an elliptic curve point (xs, ys). This may be converted into a standard key format using standard publicly known operations agreed by the nodes 3, 7. For example, the xs value may be a 256-bit integer

that could be used as a key for AES256 encryption. It could also be converted into a 160-bit integer using RIPEMD160 for any applications requiring this length key.

[0206] The common secret (CS) may be determined as required. Importantly, the first node 3 does not need to store the common secret (CS) as this can be re-determined based on the message (M). In some examples, the message(s) (M) used may be stored in data store 13, 17, 19 (or other data store) without the same level of security as required for the master private keys. In some examples, the message (M) may be publicly available.

[0207] However depending on some application, the common secret (CS) could be stored in the first data store (X) associated with the first node provided the common secret (CS) is kept as secure as the first node master private key (Vic).

[0208] Advantageously, this technique can be used to determine multiple common secrets that may correspond to multiple secure secret keys based on a single master key cryptography pair.

[0209] Hierarchy of Generator Values (keys)

[0210] For example, a series of successive Generator Values (GVs) may be determined, where each successive GV may be determined based on the preceding Generator Value (GV). For example, instead of repeating steps 310 to 370 and 410 to 470 to generate successive single-purpose keys, by prior agreement between the nodes, the previously used Generator Value (GV) can be rehashed repeatedly by both parties to establish a hierarchy of Generator Values. In effect, the Generator Value, based on the hash of a message (M), can be a next generation message (M') for the next generation of Generator Value (GV). Doing this allows successive generations of shared secrets to be calculated without the need for further protocol-establishment transmissions, in particular transmission of multiple messages for each generation of common secrets. The next generation common secret (CS') can be computed as follows.

[021 1] Firstly, both the first node 3 and the second node 7 independently determine the next generation of the Generator Value (GV). This is similar to steps 320 and 420 but adapted with the following formulas:

[0212] M' = SHA-256(M) (Equation 18)

[0213] GV = SHA-256(M') (Equation 19.1)

[0214] GV = SHA-256(SHA-256(M)) (Equation 19.2)

[0215] The first node 3 may then determine the next generation of the second node second public key (P_2s') and the first node second private key (V_2c') similar to steps 370 and 330 described above, but adapted with the following formulas:

[0216] $P_{2S}' = Pis + GV \times G$ (Equation 20. 1)

[0217] $V_{2C}' = Vic + GV$ (Equation 20.2)

[0218] The second node 7 may then determine the next generation of the first node second public key (P_2c') and the second node second private key (V_2s') similar to steps 430 and 470 described above, but adapted with the following formulas:

[0219] $P_{2C}' = Pic + GV \times G$ (Equation 21.1)

[0220] $V_{2S}' = Vis + GV$ (Equation 21.2)

[0221] The first node 3 and the second node 7 may then each determine the next generation common secret (CS'). In particular, the first node 3 determines the next generation common secret (CS') with the formula:

[0222] $CS' = V_{2C}' \times P_2s'$ (Equation 22)

[0223] The second node 7 determines the next generation common secret (CS') with the formula:

[0224] $CS' = V_{2S}' \times P2c'$ (Equation 23)

[0225] Further generations (CS'', CS''', etc.) can be calculated in the same way to create a chain hierarchy. This technique requires that both the first node 3 and the second node 7 keep track of the original Message (M) or the originally calculated Generator Value (GV), and to which node it relates. As this is publicly known information there are no security issues regarding the retention of this information. Accordingly, this information might be kept on 'hash tables' (linking hash values to public keys) and distributed freely across the network 5 (for example using Torrent). Furthermore, if any individual common secret (CS) in the hierarchy is ever compromised, this does not affect the security of any other common secrets in the hierarchy provided the private keys Vic, Vis remain secure.

[0226] Tree structure of keys

[0227] As well as a chain (linear) hierarchy as described above, a hierarchy in the form of a tree structure can be created. With a tree structure, a variety of keys for different purposes such as authentication keys, encryption keys, signing keys, payment keys, etc. may be determined whereby these keys are all linked to a single securely maintained master key. This is best illustrated in Fig. 12 that shows a tree structure 901 with a variety of different keys. Each of these can be used to create a shared secret with another party. Tree branching can be accomplished in several ways, three of which are described below.

[0228] (i) Master key spawning

[0229] In the chain hierarchy, each new 'link' (Public/Private key pair) is created by adding a multiply rehashed Message to the original master key. For example, (showing only the private key of the first node 3 for clarity):

[0230] $V_{2C} = Vic + SHA\text{-}256(M)$ (Equation 24)

[0231] $V_{2C}' = Vic + SHA\text{-}256(SHA\text{-}256(M))$ (Equation 25)

[0232] $V_{2C}'' = Vic + SHA\text{-}256(SHA\text{-}256(SHA\text{-}256(M)))$ (Equation 26)

[0233] ... and so on.

[0234] To create a branch, any key can be used as a sub-master key. For example V_2c' can be used as a sub-master key (V_3c) by adding the hash to it as is done for the regular master key:

[0235] $V_{3_C} = V_{2C}' + SHA\text{-}256(M)$ (Equation 27)

[0236] The sub-master key (V_3c) may itself have a next generation key (V_3c'), for example:

[0237] $V_{3_C}' = V_{2C}' + SHA\text{-}256(SHA\text{-}256(M))$ (Equation 28)

This provides a tree structure 903 using the master key spawning method as shown in Figure 13.

[0238] (ii) Logical Association

[0239] In this method all the nodes in the tree (public/private key pairs) are generated as a chain (or in any other way) and the logical relationships between the nodes in the tree is maintained by a table in which each node in the tree is simply associated with its parent node in the tree using a pointer. Thus the pointer may be used to determine the relevant public/private key pairs for determining the common secret key (CS) for the session.

[0240] (iii) Message Multiplicity

[0241] New private/public key pairs can be generated by introducing a new message at any point in the chain or tree. The message itself may be arbitrary or may carry some meaning or function (e.g. it might be related to a 'real' bank account number, etc). It

may be desirable that such new messages for forming the new private/public key pairs are securely retained.

CLAIMS:

1. A computer implemented method comprising:

identifying a set of first structure public keys comprising at least one public root key associated with a first structure of interest of an entity and one or more associated public sub-keys;

deriving a deterministic association between the at least one public root key and the one or more associated public sub-keys;

extracting data from a plurality of transactions (TXs) from a blockchain, the data comprising:

data indicative of a transaction (Tx) between the first structure and at least one further structure; and

a first structure public key associated with the first structure

wherein the first structure public key is part of a cryptographic public/private key;

and

generating an output for the first structure of interest by matching at least part of the set of first structure public keys to the extracted transaction data using the deterministic association.

2. A method according to claim 1 wherein:

the one or more public sub-keys is generated or determined using Elliptic Curve Cryptography (ECC) and a deterministic key (DK) that is based on a cryptographic hash of a message (M).

3. A method according to claim 1 or 2 wherein:

i) the message is selected, arranged or created so as to convey a meaning, link and/or association;

and/or

ii) the one or more public sub-keys is generated or determined based on a scalar addition of an associated public parent key and the scalar multiplication of a deterministic key (DK) and a generator (G).

4. The method of any one of the preceding claims wherein:
the step of deriving the deterministic association between the at least one public root key and the one or more associated public sub-keys may comprise determining a rule for determining the public sub-keys.

5. The method of any one of the preceding claims wherein the deterministic association between the at least one public root key and the one or more associated public sub-keys is, or is based on, a tree hierarchy.

6. The method of any one of the preceding claims wherein the step of identifying the set of first structure public keys comprises identifying the at least one root key and determining the one or more sub-keys based on the at least one root key.

7. The method of any one of the preceding claims wherein the step of extracting the transaction data from the plurality of transactions from the blockchain further comprises extracting one or more of the following data from the plurality of transactions (TXs):
a transaction (Txl) input value;
a transaction output (TXO) value;
a rule for deriving the transaction input or output value based on the data indicative of the transaction; and
a time stamp for the transaction (Tx).

8. The method of any one of the preceding claims wherein the transaction between the first structure and the further structure relates to or comprises an exchange or transfer of cryptocurrency.

9. The method of any one of claims 1 to 8, wherein the transaction between the first structure and the further structure relates to an exchange of a contract, a goods or services transaction, or a transfer of a tokenised asset.

10. The method of claim 9 wherein the transaction between the first structure and the further structure relates to an exchange of a contract, and wherein the contract confers specified rights upon the holder to be redeemed for fiat currency.

11. The method of any one of the preceding claims comprising a step of posting or transmitting the generated output to a computer based resource.

12. The method of claim 11 wherein the step of posting the generated output comprises writing the generated output to one or more posting files and signing the one or more posting files using a first structure private key, wherein the first structure private key is part of an asymmetric cryptography pair comprising the first structure private key and an associated first structure public key.

13. The method of any one of the preceding claims comprising a step of storing the generated output on an internal database or computer-based storage facility of the entity.

14. The method of any one of the preceding claims comprising a step of storing the generated output on a public database or blockchain.

15. The method of claim 14 wherein the generated output are stored on the blockchain as metadata within a blockchain transaction.

16. The method of anyone of the preceding claims comprises a step of hashing and/or signing the generated output for storing the generated output on a database or blockchain.

17. A computer-implemented method comprising the steps:

associating one or more elements of a structure with a cryptographic sub-key derived from another cryptographic key, wherein the structure is associated with, or reflective of, a hierarchical entity;
extracting or copying, from a blockchain transaction (Tx), data comprising or relating to a sub-key of an element; and
transmitting the extracted data to a non-blockchain based computer system associated with the entity.

18. A method according to claim 17 and further comprising the step of:
scanning, traversing and/or analysing the blockchain to identify one or more blockchain transactions (TXs) which comprise the data relating to or comprising a sub-key of an element.

19. A method according to claim 17 or 18 and further comprising the step of:
processing the extracted data to generate an output.

20. A method according to claim 19 and further comprising the step of:
communicating the output to a destination.

21. A computer system arranged to implement the method of any preceding claim.

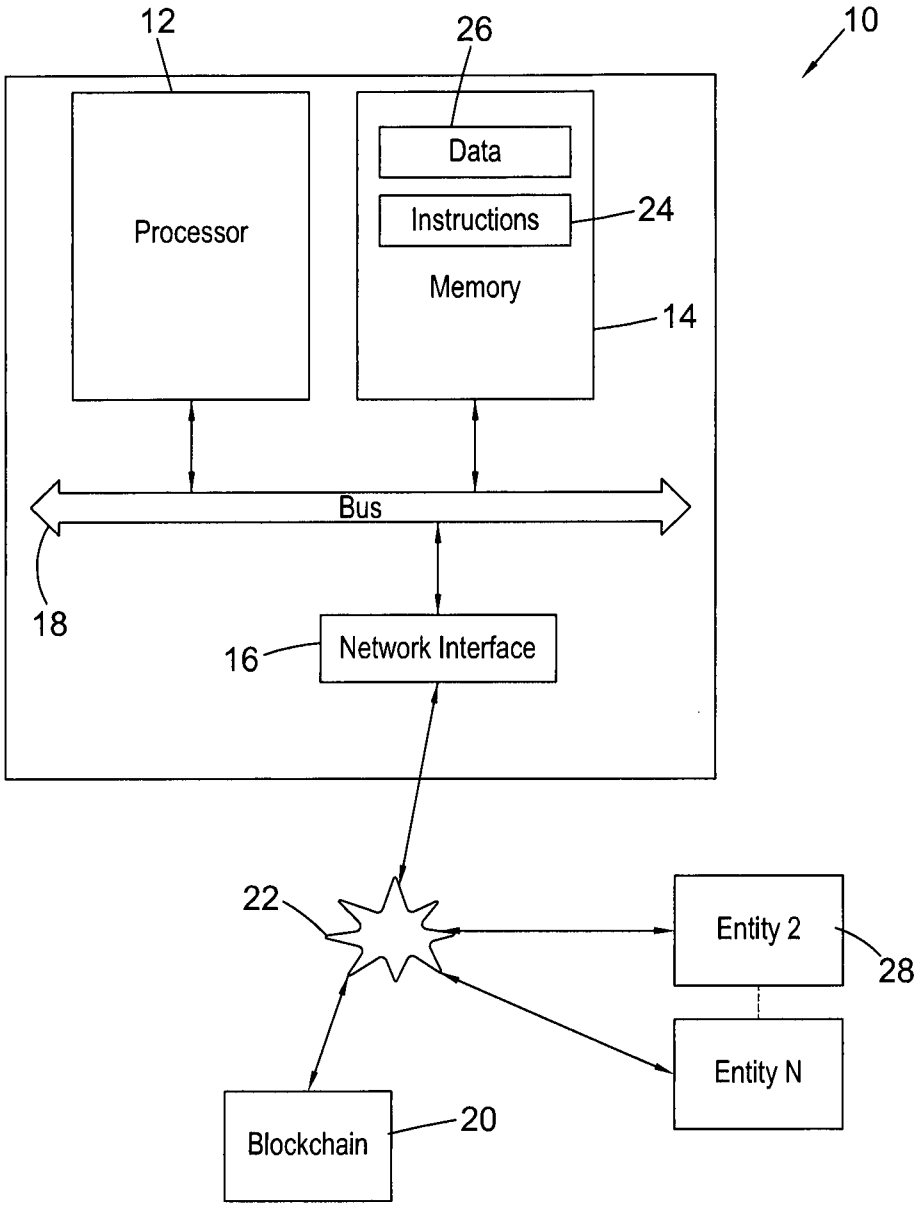
12
26
10
Processor
Data
Instructions
24
Memory
14
Bus
18
16
Network Interface
22
Entity 2
28
Entity N
Blockchain
20

Fig. 1

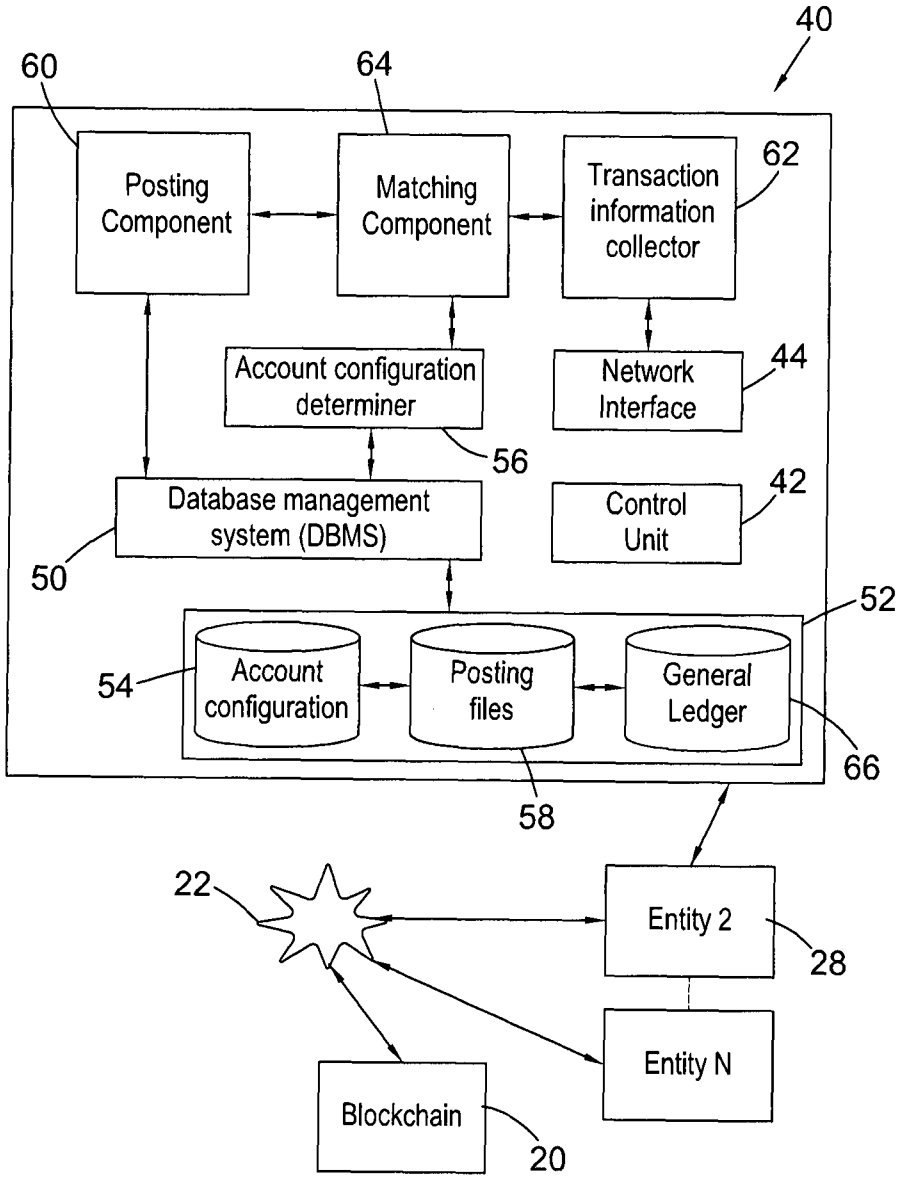
40
60
64
Posting Component
Matching Component
Transaction information collector
62
Account configuration determiner
Network Interface
44
56
Database management system (DBMS)
Control Unit
42
50
52
54
Account configuration
Posting files
General Ledger
66
58
22
Entity 2
28
Entity N
Blockchain
20

Fig. 2

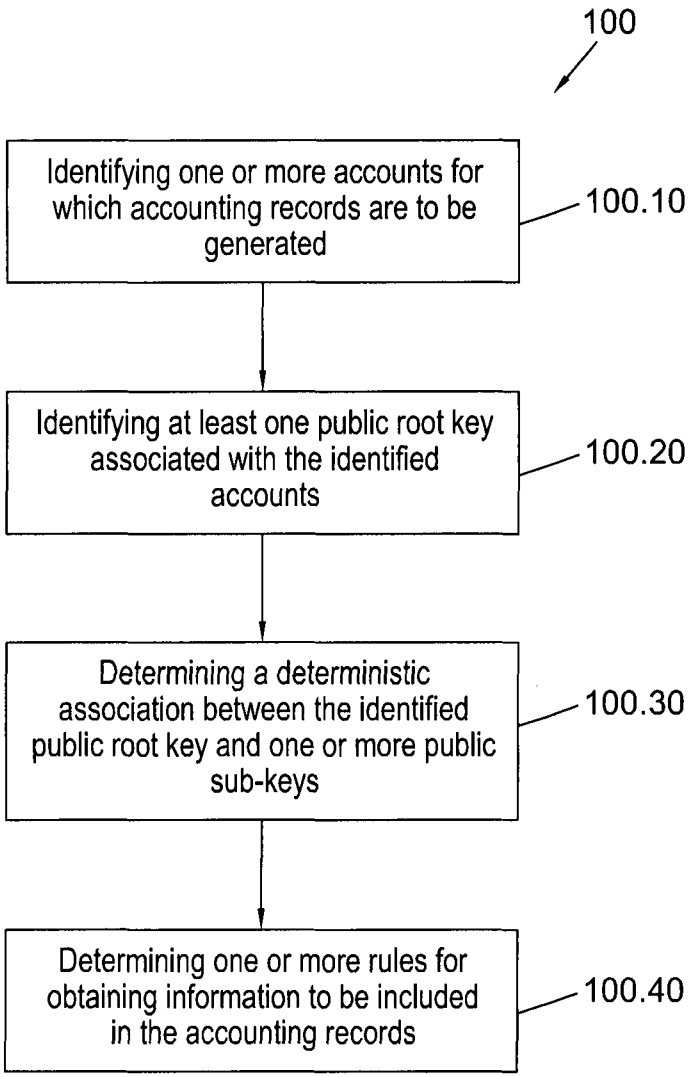
100
Identifying one or more accounts for which accounting records are to be generated
100.10
Identifying at least one public root key associated with the identified accounts
100.20
Determining a deterministic association between the identified public root key and one or more public sub-keys
100.30
Determining one or more rules for obtaining information to be included in the accounting records
100.40

Fig. 3

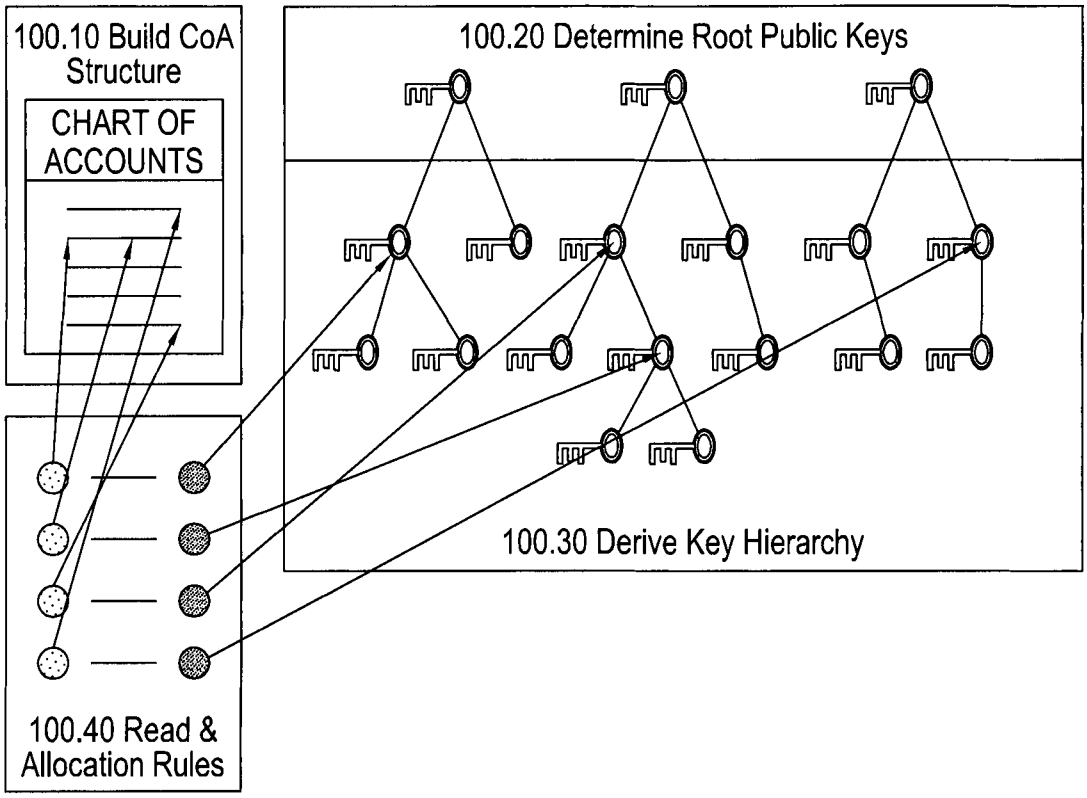
100.10 Build CoA Structure
CHART OF ACCOUNTS
100.20 Determine Root Public Keys
100.30 Derive Key Hierarchy
100.40 Read & Allocation Rules

Fig. 4

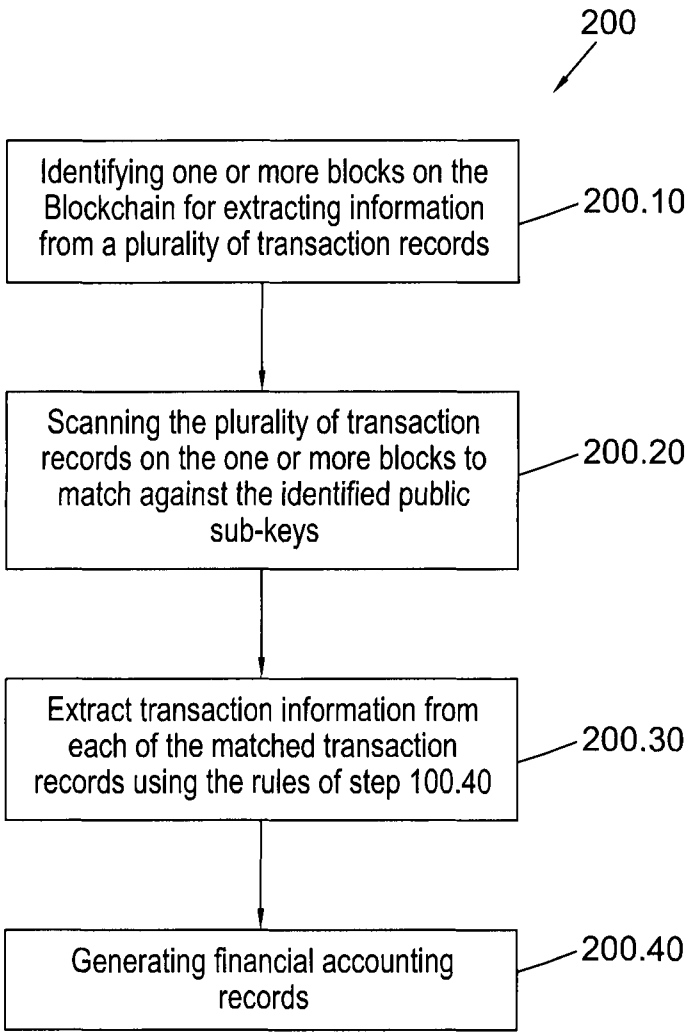
200
Identifying one or more blocks on the Blockchain for extracting information from a plurality of transaction records
200.10
Scanning the plurality of transaction records on the one or more blocks to match against the identified public sub-keys
200.20
Extract transaction information from each of the matched transaction records using the rules of step 100.40
200.30
Generating financial accounting records
200.40

Fig. 5

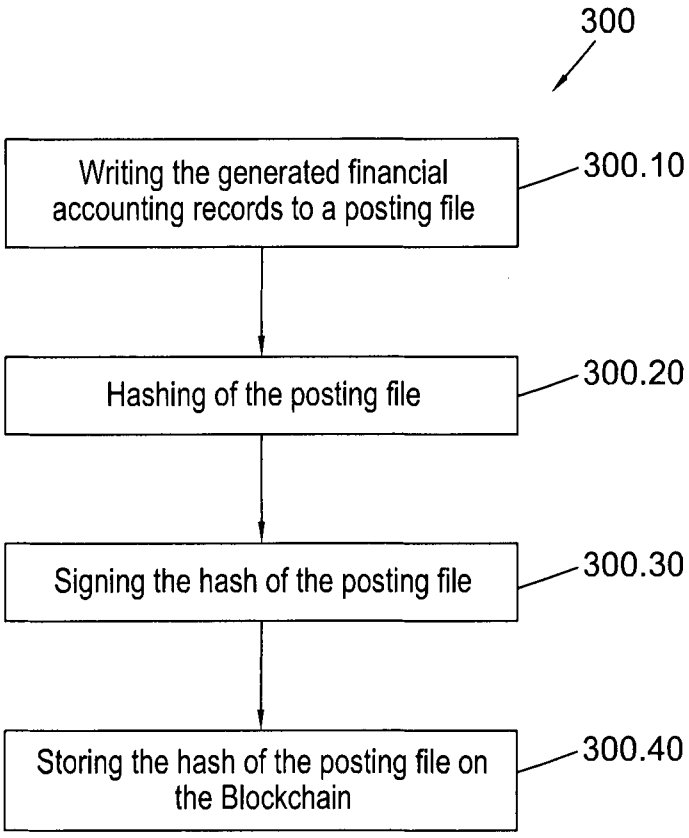
300
Writing the generated financial accounting records to a posting file
300.10
Hashing of the posting file
300.20
Signing the hash of the posting file
300.30
Storing the hash of the posting file on the Blockchain
300.40

Fig. 6

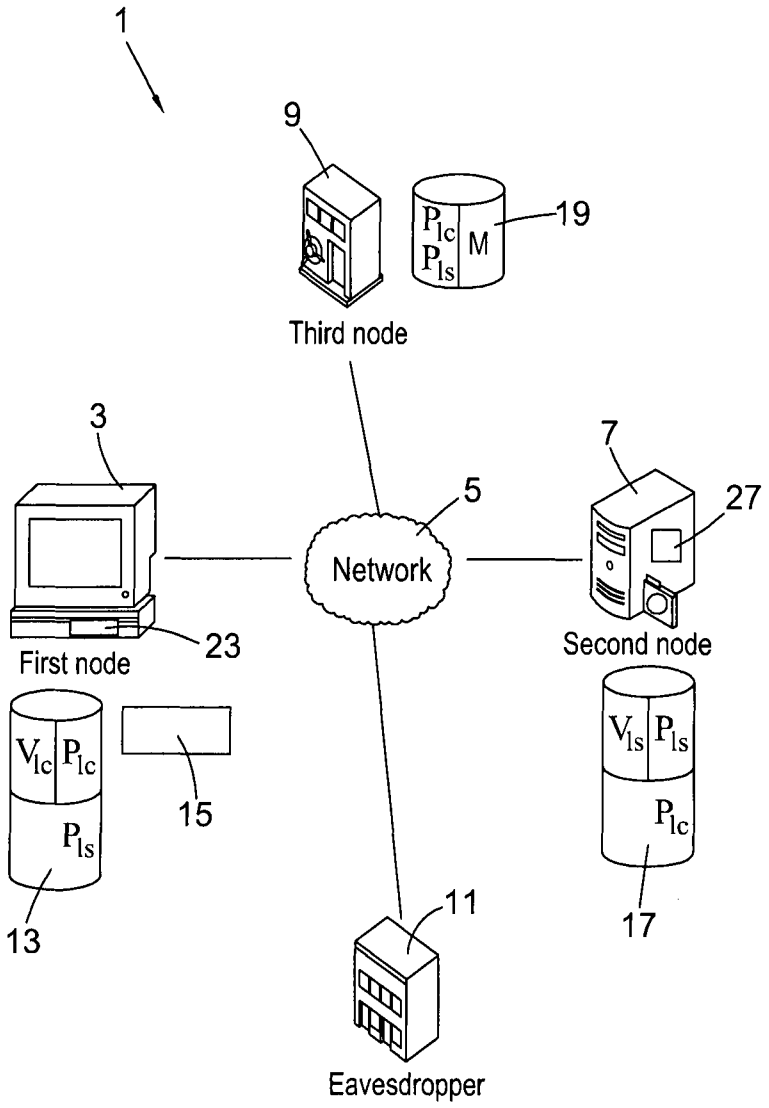
1
9
Third node
P_{lc} M
P_{ls}
19
3
First node
23
5
Network
7
Second node
27
V_{lc} P_{lc}
P_{ls}
13
15
V_{ls} P_{ls}
P_{lc}
17
11
Eavesdropper

Fig. 7

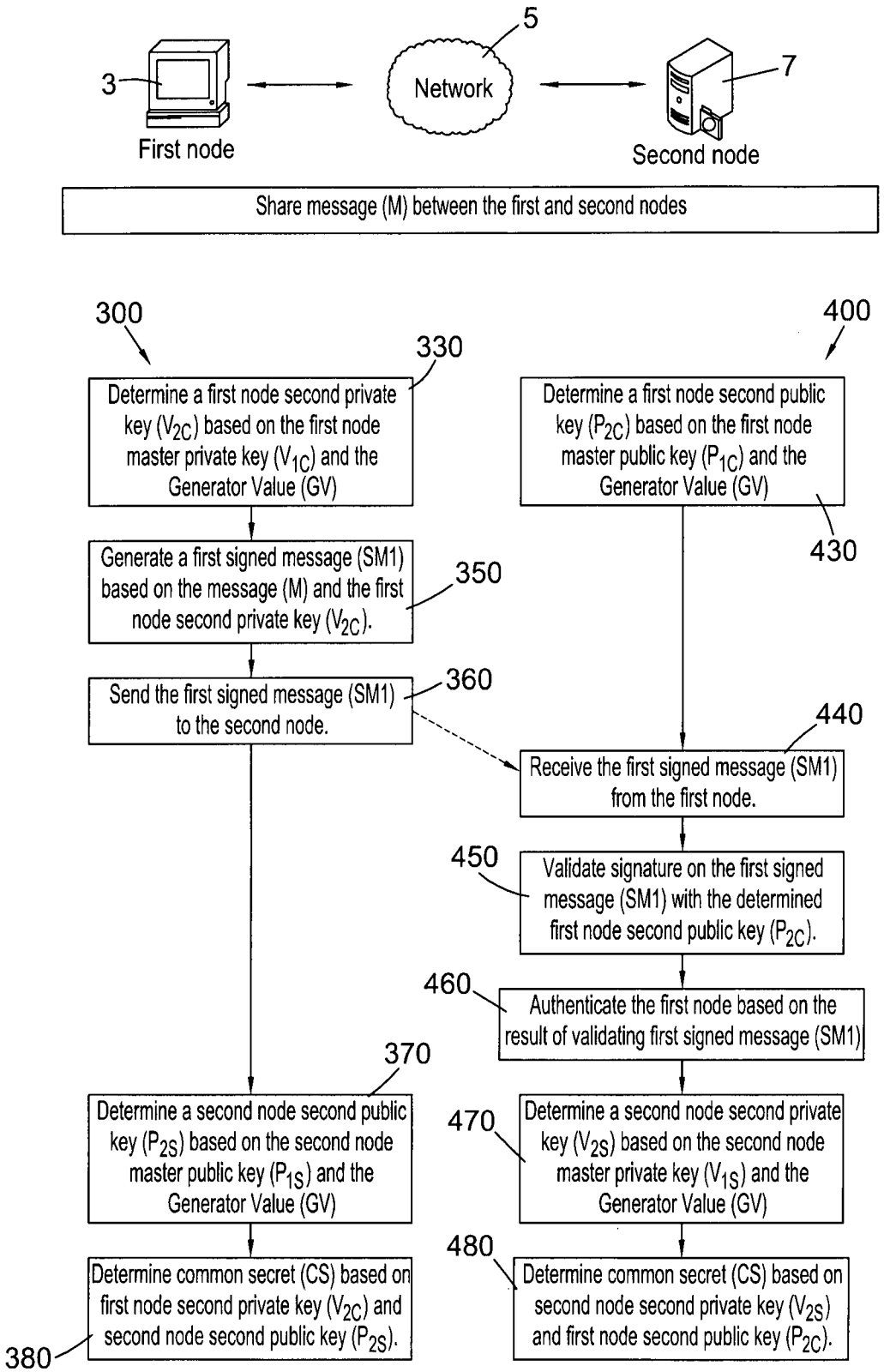
5
3
Network
7
First node
Second node
Share message (M) between the first and second nodes
300
400
330
Determine a first node second private key (V_{2C}) based on the first node master private key (V_{1C}) and the Generator Value (GV)
Determine a first node second public key (P_{2C}) based on the first node master public key (P_{1C}) and the Generator Value (GV)
430
Generate a first signed message (SM1) based on the message (M) and the first node second private key (V_{2C}).
350
Send the first signed message (SM1) to the second node.
360
440
Receive the first signed message (SM1) from the first node.
450
Validate signature on the first signed message (SM1) with the determined first node second public key (P_{2C}).
460
Authenticate the first node based on the result of validating first signed message (SM1)
370
Determine a second node second public key (P_{2S}) based on the second node master public key (P_{1S}) and the Generator Value (GV)
470
Determine a second node second private key (V_{2S}) based on the second node master private key (V_{1S}) and the Generator Value (GV)
480
Determine common secret (CS) based on first node second private key (V_{2C}) and second node second public key (P_{2S}).
Determine common secret (CS) based on second node second private key (V_{2S}) and first node second public key (P_{2C}).
380

Fig. 8

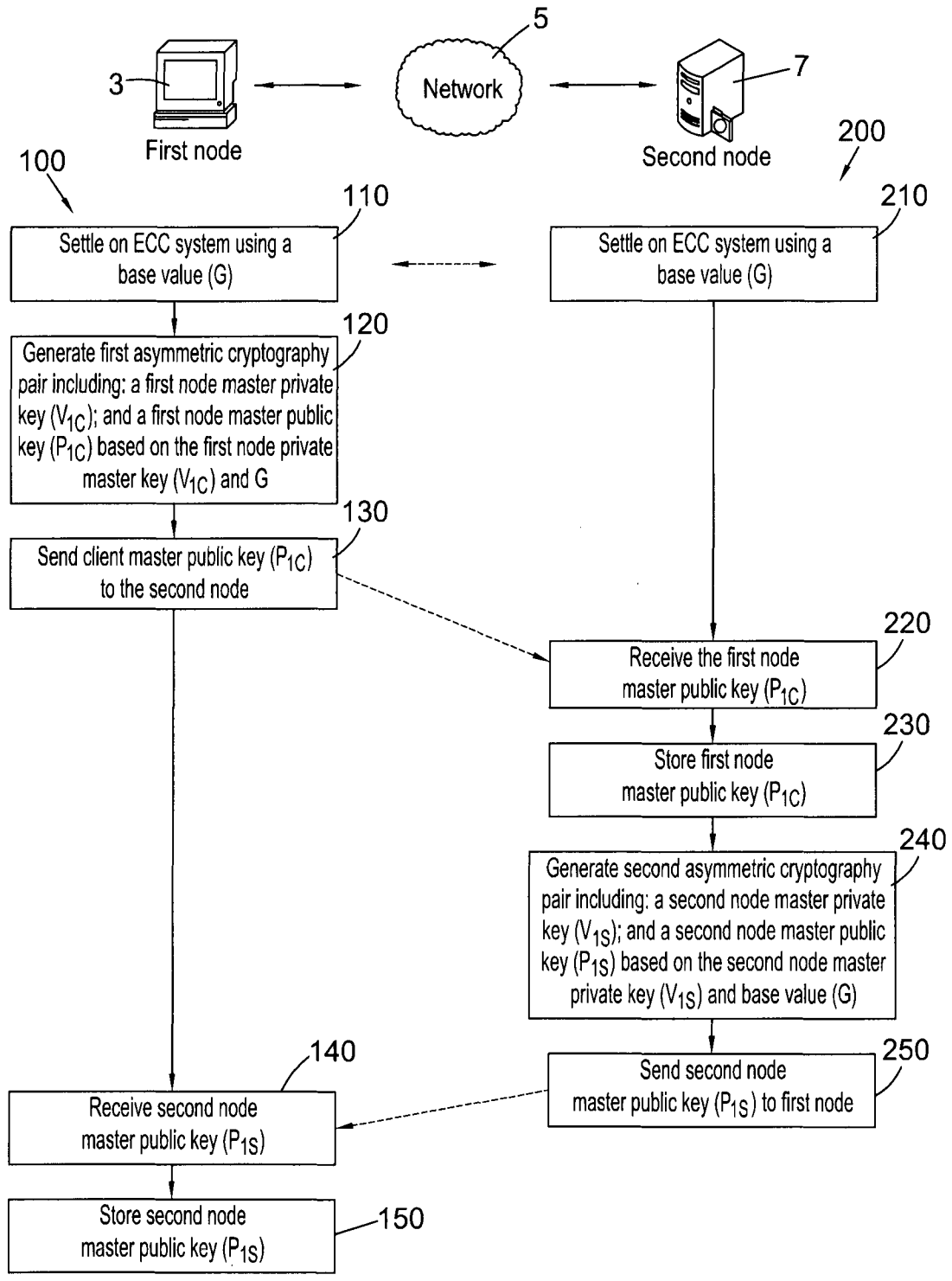
3
First node
5
Network
7
Second node
100
200
110
Settle on ECC system using a base value (G)
210
Settle on ECC system using a base value (G)
120
Generate first asymmetric cryptography pair including: a first node master private key (V_{1C}); and a first node master public key (P_{1C}) based on the first node private master key (V_{1C}) and G
130
Send client master public key (P_{1C}) to the second node
220
Receive the first node master public key (P_{1C})
230
Store first node master public key (P_{1C})
240
Generate second asymmetric cryptography pair including: a second node master private key (V_{1S}); and a second node master public key (P_{1S}) based on the second node master private key (V_{1S}) and base value (G)
250
Send second node master public key (P_{1S}) to first node
140
Receive second node master public key (P_{1S})
Store second node master public key (P_{1S})
150

Fig. 9

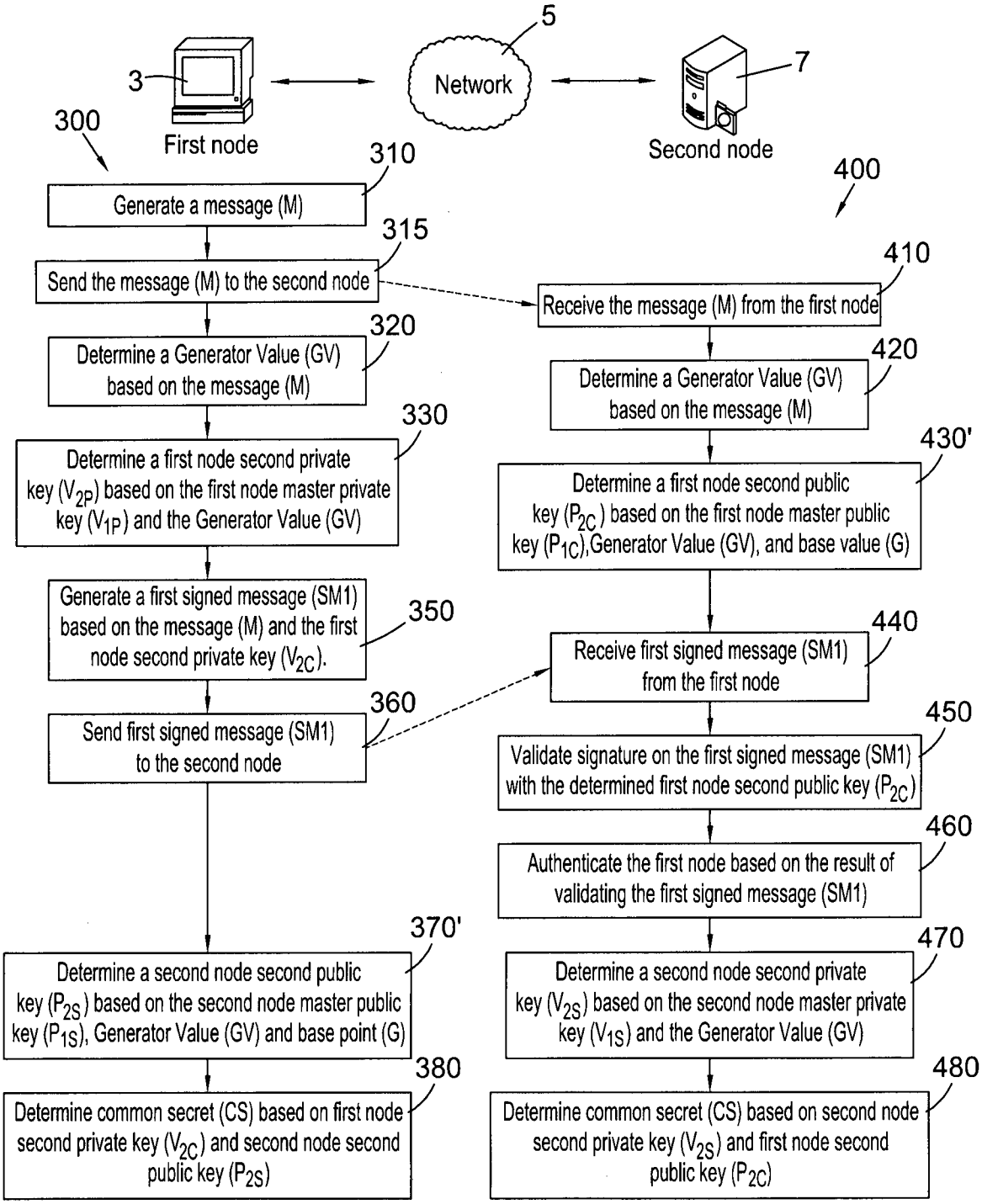
5
Network
3
7
First node
Second node
300
400
310
Generate a message (M)
315
Send the message (M) to the second node
410
Receive the message (M) from the first node
320
Determine a Generator Value (GV) based on the message (M)
420
Determine a Generator Value (GV) based on the message (M)
330
Determine a first node second private key (V_{2P}) based on the first node master private key (V_{1P}) and the Generator Value (GV)
430'
Determine a first node second public key (P_{2C}) based on the first node master public key (P_{1C}),Generator Value (GV), and base value (G)
350
Generate a first signed message (SM1) based on the message (M) and the first node second private key (V_{2C}).
440
Receive first signed message (SM1) from the first node
360
Send first signed message (SM1) to the second node
450
Validate signature on the first signed message (SM1) with the determined first node second public key (P_{2C})
460
Authenticate the first node based on the result of validating the first signed message (SM1)
370'
Determine a second node second public key (P_{2S}) based on the second node master public key (P_{1S}), Generator Value (GV) and base point (G)
470
Determine a second node second private key (V_{2S}) based on the second node master private key (V_{1S}) and the Generator Value (GV)
380
Determine common secret (CS) based on first node second private key (V_{2C}) and second node second public key (P_{2S})
480
Determine common secret (CS) based on second node second private key (V_{2S}) and first node second public key (P_{2C})

Fig. 10

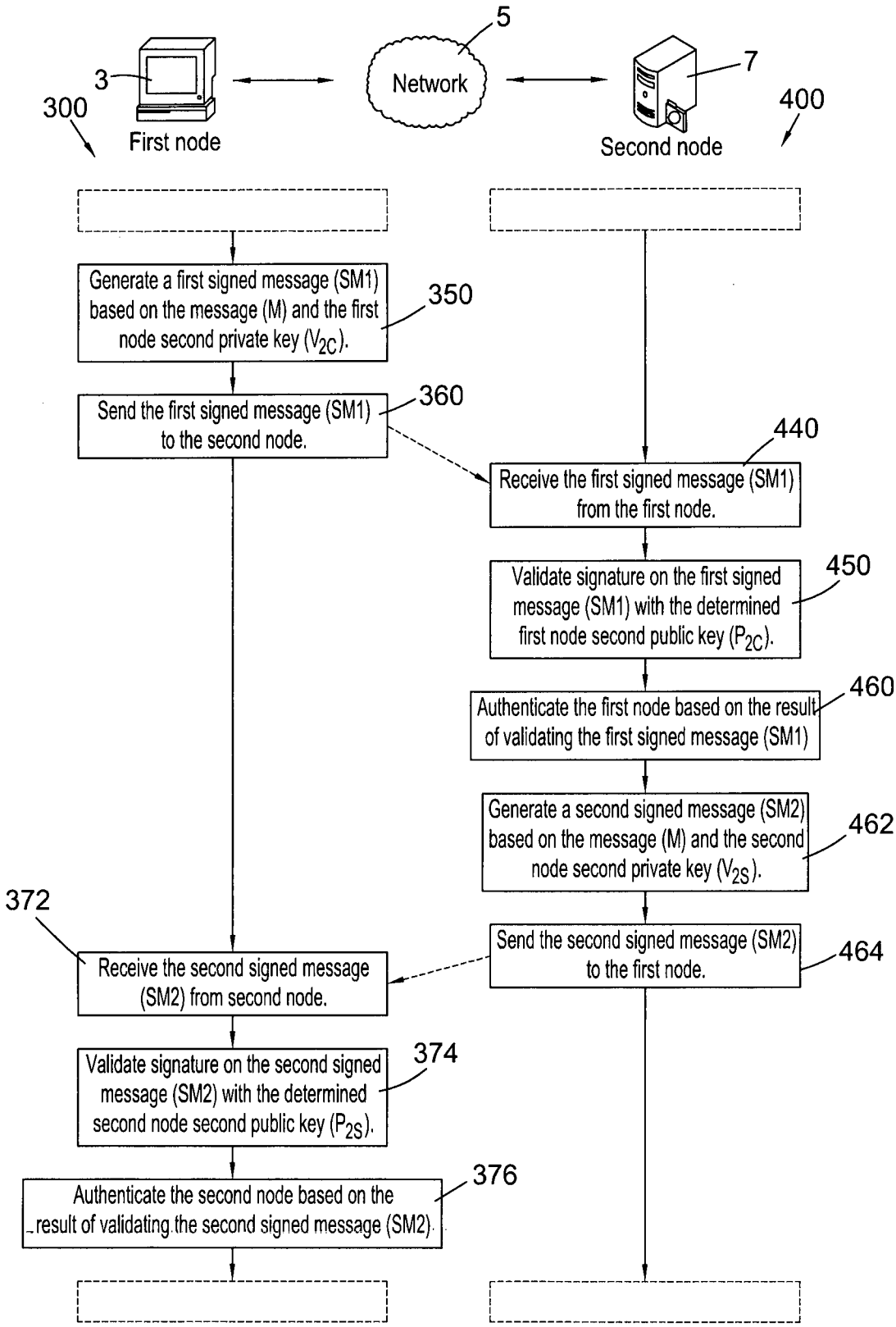
5
Network
3
300
First node
7
400
Second node
Generate a first signed message (SM1) based on the message (M) and the first node second private key (V_{2C}).
350
Send the first signed message (SM1) to the second node.
360
440
Receive the first signed message (SM1) from the first node.
Validate signature on the first signed message (SM1) with the determined first node second public key (P_{2C}).
450
Authenticate the first node based on the result of validating the first signed message (SM1)
460
Generate a second signed message (SM2) based on the message (M) and the second node second private key (V_{2S}).
462
372
Receive the second signed message (SM2) from second node.
Send the second signed message (SM2) to the first node.
464
Validate signature on the second signed message (SM2) with the determined second node second public key (P_{2S}).
374
Authenticate the second node based on the result of validating the second signed message (SM2)
376

Fig. 11

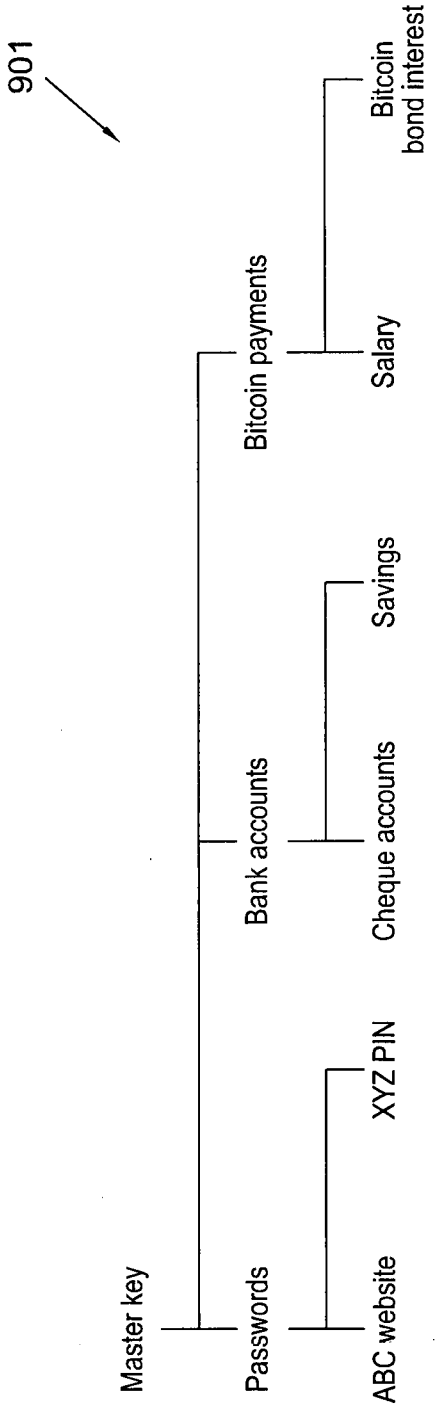
901
Master key
Passwords
ABC website
XYZ PIN
Bank accounts
Cheque accounts
Savings
Bitcoin payments
Salary
Bitcoin bond interest

Fig. 12

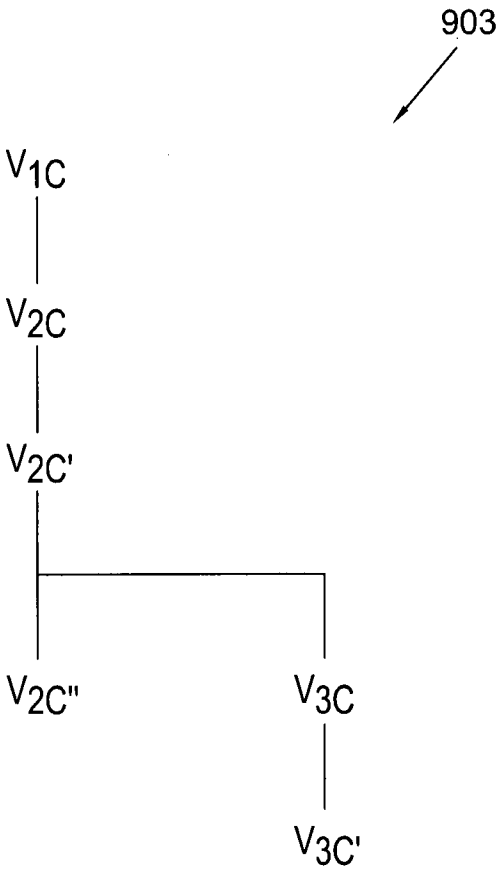

Fig. 13

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2017/05Q979

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L9/Q8 H04L9/30
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal , WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	GUY ZYSKIND ET AL: "Decentral izing Privacy: Using Bl ockchain t o Protect Personal Data" , THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC. (I EEE) CONFERENCE PROCEEDINGS, 1 May 2015 (2015-05-01) , page 180, XP055360065, Pi scataway abstract page 2, l eft-hand col umn, l ine 32 - page 4, left-hand col umn, l ine 57; tabl es 1-4 ----- -/ - -	1-21

X Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) orwhich is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 May 2017	31/05/2017

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Spranger, Stephanie

3

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Wui l i e Peter: "Hi erarchi cal Determini sti c Wal lets" , , 12 February 2016 (2016-02-12) , XP055358026, Retri eved from the Internet: URL:https ://gi thub.com/bi tcoi n/bi ps/bl ob/a b90b5289f0356282397fa9b8aa47d2238a7b380/bi p-0Q32.mediawi ki [retri eved on 2017-03-23] the whole document	1-21
A	Gus Gutoski ET AL: "Hierarchi cal determini sti c Bi tcoin wal l ets that tolerate key l eakage (Short paper) ", , 1 February 2012 (2012-02-01) , XP055369870, Retri eved from the Internet: URL: https ://eprint. iacr. org/2014/998, .pdf [retri eved on 2017-05-05] page 1, l ine 19 - page 3, l ine 22 page 4, l ine 30 - page 6, l ine 4	1-21
A	Andreas M. Antonopoul os: "Masteri ng Bi tcoi n - Unl ocki ng Di gi tal Cryptocurrenci es" In: "Mastering bi tcoi n : [unl ocki ng di gi tal cryptocurrenci es] ", 20 December 2014 (2014-12-20) , O' Rei l l y Media, Beij ing Cambri dge Farnham Köl n Sebastopol Tokyo, XP055306939 , ISBN : 978-1-4493-7404-4 page 61, l ine 1 - page 138, l ine 13	1-21

3