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

(54) Title: SECURE MULTIPARTY LOSS RESISTANT STORAGE AND TRANSFER OF CRYPTOGRAPHIC KEYS FOR BLOCKCHAIN BASED SYSTEMS IN CONJUNCTION WITH A WALLET MANAGEMENT SYSTEM

Fig. 1

(57) Abstract: The invention provides a computer- implemented solution for controlling access to a computer-related resource such as, for example, a digital wallet. In one or more embodiments, the wallet may be implemented using a blockchain such as the Bitcoin blockchain but the invention is not limited in this regard. Use of the invention during the initial set-up of the wallet can enable subsequent operations such as wallet transactions to be handled in a secure manner over an insecure channel such as the internet. A method according to an embodiment of the invention can comprise the steps of splitting a verification element (such as a private key in an asymmetric cryptography pair) into a plurality of shares; determining a common secret at two or more nodes in a network; and using the common secret to transmit at least one share of the verification element between the two or more nodes. The shares can be split such that no share on its own is sufficient to arrive at the verification element. This means that no one party stores the entire private key, providing for enhanced security of the key. Two or more shares are required to restore the key. The shares are stored at separate locations one of which is an independent back-up or safe-storage location. If one of the other shares becomes unavailable, the share can be retrieved from back up to ensure that the key (and thus the controlled resource) is still accessible. To

ensure safe transmission of the share(s), the common secret is generated at two different nodes independently of each other and then used to generate an encryption key. The encryption key can be used to encrypt at least one share of the verification element, or a message comprising it, to ensure that the share(s) are transmitted securely.

Secure Multiparty loss resistant Storage and Transfer of Cryptographic Keys for blockchain based systems in conjunction with a wallet management system

This invention relates generally to computer and data security, and more particularly to secure handling of highly sensitive data items such as cryptographic keys. . The invention provides an access control mechanism. The invention is particularly suited for, but not limited to, use with digital (software) wallets. This may include, for example, wallets used in relation to cryptocurrencies such as Bitcoin. The invention provides an advantageous access control mechanism.

Cryptography involves techniques for secure storage of sensitive data as well as its communication between two or more nodes in a network. A node may include a mobile communication device, a tablet computer, a laptop computer, desktop, other forms of computing devices and communication devices, a server device in a network, a client device in a network, one or more nodes in a distributed network, etc. The nodes may be associated with, for example, a natural person, a group of people such as employees of a company, a system such as a banking system, or a distributed, peer-to-peer ledger (i.e. blockchain).

Two or more nodes may be linked by a communications network that is unsecure and vulnerable to eavesdropping or interception by unauthorised third parties. Therefore, messages sent between nodes are often sent in encrypted form. Upon receipt, the intended recipient decrypts the messages with corresponding decryption key(s) or other decryption methods. Thus the security of such communication may be dependent on preventing the third party from determining the corresponding decryption key.

One known cryptographic method includes using symmetric-key algorithms. The keys are symmetric in the sense that the same symmetric-key is used for both encryption of a plain text message and decryption of the cipher text message. However, the symmetric-key must be transmitted to both nodes in a secure way to prevent unauthorised access to it. This may include, for example, physically delivering the symmetric-key to the (authorised) nodes so that the symmetric-key is never transmitted over an unsecure communications

network. However, physical delivery in not always a practical option. Therefore, a problem in such cryptographic systems is the establishment of the symmetric-key (which may be based on a common secret) between the nodes across an unsecure electronic network such as the internet. Thus this step of providing a symmetrical key (the common

5   secret) is a potentially catastrophic vulnerability. As the symmetric-key algorithms and protocols are simple and widely used, there is a need for two nodes to determine a common secret based symmetrical key securely across an unsecure network.

The use of asymmetric-keys, also known as public-key cryptography, alleviates this issue

10   to some extent. While the private key is kept secret its corresponding public key may be made publicly available. Its interception on a network is not catastrophic. Existing protocols include the Diffie-Hellman Key Exchange and the Three Pass Protocol.

However, storage of the private key gives rise to significant security concerns. Consider,

15   for example, a digital wallet such as a Bitcoin wallet. Digital wallets comprise software which enables a user to connect with other nodes so as to perform transactions with their electronic assets e.g. using bitcoin funds to purchase goods and services. Public-key cryptography is often used to protect the vital information which is needed for such connections and transactions. The private key is stored either by the wallet installed on

20   the user's device ('client side') or by a wallet service provider ('server side'). However, if the private key is stored only at the client side, the private key can be lost through theft, loss or damage caused to the user's hardware e.g. computer, mobile phone etc. Similarly, if the user dies or becomes incapacitated, knowledge of or access to the private key can be lost and thus the funds associated with the wallet become inaccessible. While server-side

25   storage of the private key can overcome these problems, the user must be prepared to trust the service provider to keep their private key secure. Security breaches at the server side are a real and significant risk.

Thus, it is desirable to provide a solution which enables the safe handling of a secret. This

30   secret may be a cryptographic key and/or something which may provide access to the key. Such an improved solution has now been devised. In accordance with the present invention there is provided an improved solution as defined in the appended claims.

-3-

The invention may provide a computer-implemented method. It may enable the control of access to a resource. It may be called a verification or authentication method. It may be referred to as a cryptographic key management solution. The resource may be any type of physical or electronic resource. In one embodiment, the resource is a digital wallet or some other resource relating to a form of currency. It may be a Bitcoin wallet or other wallet for the management of cryptocurrency resources. The invention may provide a method of controlling access to a digital wallet (and corresponding system).

The invention may be used during the set-up, registration or creation of a digital wallet via an unsecure communication channel (such as the internet), to enable subsequent wallet-related operations such as transactions to be handled, communicated and/or created in a secure fashion.

One or more embodiments of the invention may comprise the step of deriving the cryptographic key from an existing cryptographic key pair. This may comprise the steps of:

    determining a first entity second private key based on at least a first entity master private key and a generator value;
    determining a second entity second private key based on at least a second entity master private key and the generator value;
    determining a common secret (CS) at the first entity based on the first entity second private key and the second entity second public key, and determining the common secret (CS) at the second entity based on the second entity second private key and first entity second public key; and
    wherein:
        the first entity second public key and the second entity second public key are respectively based on at least the first/second entity master key and the generator value.

Additionally or alternatively, the invention may comprise a method of controlling access to a digital wallet, the method comprising the steps:

determining a first entity second private key based on at least a first entity master
        private key and a generator value;

determining a second entity second private key based on at least a second entity master
        private key and the generator value;

5   determining a common secret (CS) at the first entity based on the first entity second
        private key and the second entity second public key, and determining the common
        secret (CS) at the second entity based on the second entity second private key and
        first entity second public key; and

wherein:

10   the first entity second public key and the second entity second public key are
            respectively based on at least the first/second entity master key and the generator
            value.

Additionally or alternatively, the method may comprise the steps:

15   splitting a verification element into a plurality of shares;
          determining a common secret at or on two or more nodes in a network;
          using the common secret to transmit at least one share of the verification element
            from one node in the network to at least one other node.

20   The verification element may be a cryptographic key. It may be a private key in an
asymmetric cryptography pair. Additionally or alternatively, it may be a representation of
a cryptographic key, or some item which may be used to access, calculate, derive or
retrieve a cryptographic key. It may be some secret or value which can be used in a
verification process such as, for example, a mnemonic or a seed.

25

Thus, one aspect of the invention may relate to splitting a secret such as a private key into
(unique) shares. The verification element may be split into a plurality of shares such that
the verification element can be restored or regenerated from two or more of the shares.
Shamir's Secret Sharing Scheme may be used to split the verification element into shares.

30

The shares may be split such that any share on its own is of no value, meaning that it
cannot be used to arrive at the (original, un-split) verification element. The split may be

performed such that the verification element can only be restored upon combination of a predetermined number of shares. In one embodiment, any two shares may be sufficient for restoration of the verification element.

5   Another aspect of the invention may relate to safe handling or storage of the respective shares. The shares may be sent to, and stored by, different parties. Some or all of these parties may be nodes on the network. In one embodiment, the method may comprise the step of storing at least three shares of the verification element at different locations relative to each other.

10

At least one of the shares may be stored in or on a back-up or "safe-storage" facility. This may be separate, independent and/or distinct from any other location which stores a share. This provides an important advantage, because it enables restoration of the verification element in the event that one of the other shares becomes unavailable. In such a situation,

15   the share may be retrieved from the safe-storage facility.

A verification process may be performed prior to restoration of the verification element using the shares. The verification process may comprise verification of the identity of a pre-determined or designated individual, and/or a computing resource.

20

Another aspect of the invention may relate to the secure distribution of one or more of the shares. The method may comprise the step of using the common secret to generate an encryption key, wherein the encryption key is used to encrypt at least one share of the verification element or a message comprising said at least one share.

25

The common secret may be determined at the at least two nodes independently of each other. Thus, each node may determine or generate the secret for themselves, without input from or communication with the other node or another party. This means that the common secret may not require transmission over a communications channel. This provides

30   enhanced security because it cannot be intercepted by unauthorised parties. The common secret may be common to (i.e. shared by) only the at least two nodes. The common secret may then be used to generate an encryption key, and that encryption key may be used for

-6-

the safe transmission of the share(s). Other data may also be transmitted using the encryption key.

The method may comprise the step of determining, at a first node (C), a common secret (CS) that is common with the first node (C) and a second node (S), wherein the first node (C) is associated with a first asymmetric cryptography pair having a first node master private key (Vic) and a first node master public key (Pic), and the second node (S) is associated with a second asymmetric cryptography pair having a second node master private key (Vis) and a second node master public key (Pis), wherein the method comprises:

- determining a first node second private key (V_2c) based on at least the first node master private key (Vic) and a Generator Value (GV);

- determining a second node second public key (P_2s) based on at least the second node master public key (Pis) and the Generator Value (GV); and

- determining the common secret (CS) based on the first node second private key (V_2c) and the second node second public key (P_2s),

wherein the second node (S) has the same common secret (S) based on a first node second public key (P_2c) and a second node second private key (V_2s), wherein: the first node second public key (P_2c) is based on at least the first node master public key (Pic) and the Generator Value (GV); and the second node second private key (V_2s) is based on at least the second node master private key (Vis) and the Generator Value (GV).

The Generator Value (GV) may be based on a message (M). The method may further comprise: generating a first signed message (SMI) based on the message (M) and the first node second private key (V_2c); and sending, over the communications network, the first signed message (SMI) to the second node (S), wherein the first signed message (SMI) can be validated with a first node second public key (P_2c) to authenticate the first node (C).

The method may also comprise: receiving, over the communications network, a second signed message (SM2) from the second node (S); validating the second signed message (SM2) with the second node second public key (P2S); and authenticating the second node (S) based on the result of validating the second signed message (SM2), wherein the second

-7-

signed message (SM2) was generated based on the message (M), or a second message (M2), and the second node second private key (V_2s).

The method may further comprise generating a message (M); and sending, over a communications network, the message (M) to the second node (S). Alternatively, the method may comprise receiving the message (M), over the communications network, from the second node (S). In yet another alternative, the method may comprise receiving the message (M), over the communications network, from another node. In yet another alternative, the method may comprise receiving the message (M) from a data store, and/or an input interface associated with the first node (C).

The first node master public key (Pic), second node master public key (Pis) may be based on elliptic curve point multiplication of respective first node master private key (Vic) and second node master private key (Vis) and a generator (G).

The method may further comprise the steps of: receiving, over the communications network, the second node master public key (Pis); and storing, at a data store associated with the first node (C), the second node master public key (Pis).

The method may further comprise the steps of: generating, at a first node (C), the first node master private key (Vic) and the first node master public key (Pic); sending, over the communications network, the first node master public key (Pic) to the second node (S) and/or other node; and storing, in a first data store associated with the first node (C), the first node master private key (Vic).

The method may also comprise: sending, over the communications network, to the second node, a notice indicative of using a common elliptic curve cryptography (ECC) system with a base point (G) for the method of determining a common secret (CS). The step of generating the first node master private key (Vic) and the first node master public key (Pic) may comprise: generating the first node master private key (Vic) based on a random integer in an allowable range specified in the common ECC system; and determining the

-8-

first node master public key (Pic) based on elliptic curve point multiplication of the first node master private key (Vic) and the base point (G) according to the following formula:

$$Pic = Vic \ x \ G$$

The method may further comprise: determining the Generator Value (GV)based on

5   determining a hash of the message (M), wherein the step of determining a first node second private key (V2C) is based on a scalar addition of the first node master private key (Vic) and the Generator Value (GV) according to the following formula:

$$V_2c = Vic + GV$$

10   The step of determining a second node second public key (P2S) may be based on the second node master public key (Pis) with elliptic curve point addition to the elliptic curve point multiplication of the Generator Value (GV)and the base point (G) according to the following formula:

$$P_2s = Pis + GV \ x \ G.$$

15

The Generator Value (GV)may be based on determining a hash of a previous Generator Value (GV).

The first asymmetric cryptography pair and the second asymmetric cryptography pair may

20   be based on a function of respective previous first asymmetric cryptography pair and previous second asymmetric cryptography pair.

In an alternative wording, the invention may provide a method comprising the steps:
splitting a verification element into a plurality of shares;

25   generating, at a first node, a derived (or second) private cryptographic key based on a first master asymmetric key pair;
using the derived private key for the encryption and/or secure transmission of least one share of the verification element.

30   The method may also comprise the step of generating, at a second node, the same derived private key, this being generated independently of the first node and based on a second master asymmetric key pair.

The derived private key may be part of an asymmetric key pair comprising the private key and a public key. The first and/or second nodes may use Elliptic Curve Cryptography (ECC) to generate the private key (and its corresponding public key).

The method may comprise the steps:

- Agreeing on, between the first and second nodes, a standard ECC system using a base point (G); and/or

- generating, at the first and/or second node, a public/private key pair using the agreed standard ECC system and publishing the public key; this may mean making it publicly available; and/or

- registering the first node's master public key ($P_M C$) at the second node or another location; and/or registering the second node's master public key ($P_M C$) at the first node or another location; and/or

- sending a message (M) from the first node to the second node and/or vice versa, and creating a hash of the message; the message may be signed using the derived private key; this step may represent the only transmission required to 1) establish a shared secret between the nodes and 2) initiate a secured communication session between them. The first or second node may use the received message M to generate its own derived (secondary) public/private key pair. This may allow the node to calculate the other node's derived public key; and/or

- receiving the message and independently calculating the hash of the message M (e.g. SHA-256(M)); and/or

- calculating a public key ($P_2 c$) which is derivable from the master key (PMC); and/or

- validating the signature (Sig-V $_2$c) against the calculated $P_2 c$

The derived private key may be deterministically derived from the first or second node's master public key.

The invention may also comprise a computer-implemented system arranged and configured to implement any embodiment of the method(s) described above. The

-10-

system may comprise or utilise a blockchain network or platform. Additionally or alternatively, it may comprise a digital wallet provider or management system.

Any feature described above in relation to one aspect or embodiment of the invention may

5   also be used in relation to any other aspect or embodiment. For example, and feature described in relation to the method may apply to the system and vice versa.

These and other aspects of the present invention will be apparent from and elucidated with reference to, the embodiment described herein.

10

An embodiment of the present invention will now be described, by way of example only, and with reference to the accompany drawings, in which:

Figure 1 is a schematic diagram of an example system to determine a common secret for a

15   first node and second node, as may be used in accordance with the present invention for secure transmission of highly sensitive information, such as a share of a private key;

Figure 2 is a flow chart of computer-implemented methods for determining a common secret as may be used in accordance with the present invention for secure transmission of

20   highly sensitive information, such as a share of a private key;

Figure 3 is a flow chart of computer-implemented methods to register the first and second nodes ;

25   Figure 4 is another flow chart of computer-implemented methods for determining a common secret as may be used in accordance with the present invention for secure transmission of highly sensitive information, such as a share of a private key;

Figure 5 is a flow chart of computer-implemented methods of secure communication

30   between the first node and second node.

-11-

As explained above, a need exists for enhanced storage and/or exchange of a secret such as a cryptographic key, or a secret which can be used to generate a key. The secret may be a seed for a wallet mnemonic, or other security-related item. The invention provides such a solution. An embodiment is described below for the purposes of illustration, and uses the context of a digital wallet implemented on a blockchain. However, the invention is not limited to such implementations and could be implemented in respect of any computer-implemented network or system.

As above, public-key cryptography is often used in relation to digital wallets. If the end user (which we may refer to as a "client" or simply "user") is responsible for storing their private key, problems may arise when the user or their hardware become unavailable as this renders the private key, and thus the wallet's funds, inaccessible. However, storage of the key at the wallet provider's end (which we may refer to as "server side") requires a degree of trust in that provider and their security mechanisms. So there is a need to store the private key in such a way that it cannot be obtained by an unauthorised party, but can also be reproduced when necessary. The term "user" may be a human user or a computer-implemented resource.

One known cryptographic algorithm, known as "Shamir's secret sharing scheme" (4S), teaches splitting the secret up into unique parts or shares which are then distributed to different parties. The shares can be used to reconstruct the secret thereafter. Each individual share is of no value or use on its own until it is combined with one or more other shares. The number of shares required to reconstruct the secret can vary according to the needs of the situation. In some cases, all shares may be required, while in other cases only a sufficient number are required. This is known as the threshold scheme, where any k of the shares are sufficient to reconstruct the original secret.

In this illustrative embodiment, 4S is used to split a secret such as a private key or mnemonic seed into a number of parts. It is then also used to regenerate the key or mnemonic seed from a certain number of parts. The use of mnemonics is known in conjunction with digital wallets. The mnemonic is a human-friendly code or group of words which can be turned into a binary seed for the generation of a wallet or data.

Herein, there following terms may be used.

- "Secret" (S) is a secret (e.g. a number or value) that needs to be shared securely between parties.
- "Share" is a piece of the secret. The secret is divided into pieces and each piece is called a share. It is computed from the given secret. In order to recover the secret, one must obtain a certain numbers of shares.
- "Threshold" (k) is the minimum number of shares that one needs to regenerate or recover the secret. The secret can be regenerated only when you have $>=$ k shares.
- "Prime" (p) is a random prime number.

From a broad perspective, an illustrative embodiment may comprise a method as follows. In this example, we use a '2-of-3' scheme (i.e. k = 2):

- A user registers with a wallet provider to generate and set up a new wallet associated with that user. In this example, the wallet is a Bitcoin wallet, which utilises the Blockchain
- A public-private key pair is generated and associated with the user's wallet;
- The private key is split into shares, using 4S
- One share of the private key is sent via a secure transmission to the user
- Another share of the private key is retained by the service provider and stored on a server
- Another share is sent via a secure transmission to a remote location for safe storage. The term 'remote' does not imply any particular geographical distance or location. Instead, it is used herein to mean that the share is held in, at or on a secure storage facility or resource which is independent in some sense from the wallet provider or the user, preferably both. "Independent" may include physical, logical, financial, political and/or organisationally independent. For example, the safe storage may be contracted out to a commercial entity which provides the safe storage service for a fee; or it may be held by the user's attorney, or some other elected (and trusted) party who accepts responsibility for storing the share and supplying it upon request if needed;

-13-

- The wallet provider can destroy any or all copies of the complete private key, because it is no longer needed. When the private key is needed for subsequent authorisation of the user (e.g. because the user now wishes to make a transaction) the key can be reconstructed from the user's share, which the user provides to the wallet provider as and when needed, and the wallet provider's share.

An advantage of this is that even if the wallet provider's security is breached, the unauthorised party cannot gain access to the user's private key because it is not stored anywhere on the wallet provider's system and the wallet provider's system alone does not contain enough shares to allow reconstruction of the private key. This same advantage applies in situations where the client's security is breached.

Another advantage is that by storing a share at a safe storage location, the private key can be reconstructed by retrieving that share from safe storage and combining it with the wallet provider's share. Thus, if the user dies or becomes incapacitated, or if the user's hardware (and thus share) is lost, damaged or stolen, the funds in the wallet can still be accessed. In such a situation, the user's identity would be verified. In some cases, the identity of a proven, trusted party such as executor of an estate or attorney would be verified. This may be achieved, by example, upon production of evidence such as death certificate, passport, a legal document or other form of identification. Upon verification of authorised identity, the share of the secret would be retrieved from safe storage. Therefore, the safe storage serves as a type of back-up facility which can be used in exceptional or pre-determined circumstances.

Thus, the invention provides an advantageous combination of enhanced system/data security plus convenience. It provides a simple, effective and secure solution for access control.

It should be noted that in the above example, the private key is generated by the wallet service provider and respective parts are sent to the user and safe storage resource. However, this may not be the case in other embodiments. It is also important to note that transmission of parts between parties, which may be referred to as 'nodes', must be

performed in a secure manner because any unauthorised interception of multiple shares could enable the interceptor to reconstruct the secret (e.g. mnemonic or key). This secure exchange problem is also addressed by the invention, as described below.

5   More detailed aspects of the invention are now described for the purpose of illustration. It should be noted that Shamir's Secret Sharing Scheme is a technique which is known in the art, and the skilled person would be aware of, understand and be able to use it. Therefore, the following is provided for the purpose of completeness only.

10   **Splitting the Secret into shares**
Given a secret S, a number of participants n, a threshold number k, and some prime number p, we construct a polynomial:

$$y = f(x) \text{ of degree } k-1 \text{ (modulo our prime } p)$$

with constant term *S*.

15

Next, we choose n unique random integers between 1 and p-1, inclusive, and evaluate the polynomial at those n points. Each of the n participants is given a (x, y) pair. This can be achieved by the following steps.

20   1. Convert into Integer
For the 4S algorithm, the secret needs to be an integer. Hence if the secret is in some other format (ex. String, hex etc.) it must be converted into an integer first. If the secret is already an integer, this step can be omitted. For this example, let *S = 1234*.

25   2. Decide number of shares (n) and threshold (k)
Note that *k* parts will be required to regenerate the secret. Hence, choose *S* and *k* such that *k* parts can always be obtained while recovering the secret. For this example, let *n = 6, k = 3*.

30   3. Create the polynomial
We need to create a polynomial of the form: $y = f(x) \bmod p$
    i.   Determine constant term and degree of polynomial

-15-

$$f(x) = a_o + a_ix + a_2x^2 + a_3X^3 + \ldots + a_{k-i}x^{k-1}$$

- The constant term $a_o = S$
- degree of polynomial $= k-1$

Hence for $k = 3$ and $S = 1234$, we need to build a polynomial with *degree 2* and $a_o = 1234$

$$f(x) = 1234 + a_ix + a_2x^2$$

ii.   Determine coefficients

Chose *k-1* random numbers (use a Random (or pseudo random) Number Generator) such that:

$0 < a_n < S$

Let $a_1 = 166$; $a_2 = 94$

Hence, f(x) = 1234 + 166x + 94 x^2

iii. Select a random prime number

Chose a random prime number (p) such that:

$p > max(S,n)$

Let p = 1613

iv.   Final polynomial

y = f(x) mod p

y = (1234 + 166x + 94 x^2) mod 1613

Creating the shares

To divide the secret into *n* shares, we need to construct *n* points (shares) using the polynomial:

y = (1234 + 166x + 94 x^2) mod 1613

Since *n = 6* for this example, we will have 6 points. Note that we start with *x = 1 and NOT x = 0.*

-16-

For *x = 1 to 6,* the 6 points are as follows:

(1, 1494); (2,329); (3,965); (4,176); (5, 1188); (6,775)

Out of these n (6) points, any k (3) points can be used to regenerate the secret key.

5

Reconstructing the Secret from a given number of shares

  i.   Get the secret integer

      To reconstruct the secret, we need following information:

      n = 6, k =3, p =1613,

10   *k shares:*

      (xO, yO) = (1, 1494); (xl, yl) = (2,329); (x2, y2) = (3,965)

      Once we have the above information, we can use a technique such as Lagrange
Interpolation which is known in the art and readily appreciated by the skilled person.

15   Using this technique we can rebuild the entire polynomial. The coefficients can be
calculated according to formula below:

$a_i(x) = \left[\sum^{k-1}_{i=0} y_i \prod_{0<=j<=k-i, j \neq i} (x-X_j)/(X_i - X_j) \right] \bmod p$

but since *S = ao,* we   only need to find *ao = ao*

20   *(0)*

$$a_0 = \left[\sum^{k-1}_{i=0} y_i \prod_{0<=j<=k-1, j \neq i} \frac{-x_j}{x_i - x_j} \right] \bmod p$$

where $x_i - x_j \neq O$

25 The skilled person will understand that in the above, the exponent - 1 signifies taking the
multiplicative inverse. Most programming languages comprise inbuilt packages to perform
mathematical operations such as multiplicative inverse.

  ii.   Convert integer to desired format

30   If Step 1 was performed to convert a specific format to an integer, we follow the
reverse procedure to convert the integer back to the desired format.

-17-

Secure transmission of the shares

As mentioned above, it is important that the shares of the secret are transmitted to the respective recipients in a secure manner so as to prevent unauthorised parties from being able to reconstruct the secret. In a preferred embodiment, the secure transmission can be
5   achieved as described below.

A common secret (CS) can be established between two parties and then used to generate a secure encryption key for transmission of one or more of the shares. This common secret (CS) is not to be confused with the secret (S) referred to above. The Common Secret (CS)
10 is generated and used to enable secure exchange of the Secret (S) e.g. key or share thereof.

The two parties could be any two of the wallet service provider, the user, the safe storage resource or some other legitimate party. Hereafter, for the sake of convenience, they will be referred to as a first node (C) a second node (S). The aim is to generate a common (CS)
15 secret which both nodes know but without that common secret having been sent via a communication channel, thus eliminating the possibility of its unauthorised discovery. The secret splitting plus safe storage technique, in combination with a secure transmission technique such as described below, provides a secure key-management solution.

20 The secure transmission technique of the present invention involves the CS being generated at each end of the transmission in an independent manner, so that while both nodes know the CS it has not had to travel over potentially unsecure communication channels. Once that CS has been established at both ends, it can be used to generate a secure encryption key that both nodes can use for communication thereafter. This is of
25 particular benefit during the wallet registration process, for transmission of the split private key from one party to another.

Figure 1 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
30 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

-18-

example, the first node 3 may be a client (user) device and the second node 7 may be a server. The server may be a digital wallet provider's server.

The first node 3 is associated with a first asymmetric cryptography pair having a first node

5   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.

10   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.

15   To determine the 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.

The method 300 performed by the first node 3 includes determining 330 a first node

20   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

25   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).

Importantly, the same common secret (CS) can also be determined at the second node 7 by

30   method 400. The method 400 includes determining 430 a first node second public key (P2c) 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

-19-

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).

5 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

10 determine a common secret without transmitting the common secret over the communications network 5.

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

15 unsecure communications network 5. In turn, the common secret may be used as a secret key (or as the basis of a secret key) for encrypted communication between the first and second nodes 3, 7 over the communications network 5.

The methods 300, 400 may include additional steps. The method 300 may include, at the

20 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)

25 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

30 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

-20-

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.

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 yet another example, the message may be generated at a third node 9 and the message sent to both the first and second nodes 3, 7. In yet another alternative, a user may enter the message through a user interface 15 to be received by the first and second nodes 3, 7. In yet another example, the message (M) may be retrieved from a data store 19 and sent to the first and second nodes 3, 7. In some examples, the message (M) may be public and therefore may be transmitted over an unsecure network 5.

In further examples, one or more messages (M) may be stored in a data store 13, 17, 19, where the message may be associated with some entity such as digital wallet, or a communication session established between the first node 3 and the second node 7. Thus the messages (M) may be retrieved and used to recreate, at the respective first and second nodes 3, 7, the common secret (CS) associated with that wallet or session.

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. This may be advantageous if numerous transactions are performed at the first and second nodes 3, 7 and it would be impractical to store all the messages (M) at the nodes themselves.

Method of registration 100, 200

An example of a method of registration 100, 200 will be described with reference to Figure 3, 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.

-21-

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.

5

Standards for ECC may include known standards such as those described by the Standards for Efficient Cryptography Group (vvwvv.sceg.org). Elliptic curve cryptography is also described in US 5,600,725, US 5,761,305, US 5889,865, US 5,896,455, US 5,933,504, US 6,122,736, US6,141,420, US 6,618,483, US 6,704,870, US 6,785,813, US 6,078,667, US

10   6,792,530.

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

15   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.

Turning now to the first node 3, the method 100 includes settling 110 on the common ECC

20   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

25   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).

30   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,

-22-

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:

$$Pic = Vic \times G \qquad\qquad (Equation~1)$$

Thus the first asymmetric cryptography pair includes:

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

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

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.

The method 100 further includes sending 130 the first node master public key (Pic), over the communications network 5, to the second node 7. 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.

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:

$$Pis = Vis \times G \qquad\qquad (Equation~2)$$

Thus the second asymmetric cryptography pair includes:

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

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

-23-

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).

5

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

10   requested and received by the second node 7 only when determining the common secret (CS) is required (and vice versa).

The registration steps may only need to occur once as an initial setup of, for example, the digital wallet.

15

Session initiation and determining the common secret by the first node 3

An example of determining a common secret (CS) will now be described with reference to Figure 4. 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

20   desirable, or secure, to use the same common secret (CS). Thus the common secret (CS) may be changed between different sessions, time, transactions, etc.

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

25

Generating a message (M) 310

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

30   example, the message (M) may be provided as:

   Message (M) = UnixTime + nonce   (Equation 3)

-24-

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

5   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)32*0*

10   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:

$$GV = SHA\text{-}256(M) \qquad\qquad\qquad (Equation\ 4)$$

15

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
20   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

25   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:

$$V_{2C} = Vic + GV \qquad\qquad\qquad (Equation\ 5)$$

30

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:

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

Substitution of V_2c from Equation 5 into Equation 6 provides:

$$Pic = (Vic + GV) \times G \qquad \text{(Equation 7)}$$

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

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

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

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

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

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

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.

-26-

The first signed message (SMI) can be verified with the corresponding first node second public key (P_2c) 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.

5

Determine a second node second public key 370'

The first node 3 may then determine 370 a second node second public key (P_2s). As discussed above, the second node second public key (P_2s) 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:

$$P_2s = Vis \times G \qquad\qquad (\text{Equation } 10.1)$$
$$Pis = Pis + GV \times G \qquad\qquad (\text{Equation } 10.2)$$

15

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.

20

Determine the common secret 380 at the first node 3

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:

$$S = Vic \times Pis \qquad\qquad (\text{Equation } 11)$$

Method 400 performed at the second node 7

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.

-27-

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.

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:

$$P_2c = Vic \ x \ G \hspace{4cm} \text{(Equation 12.1)}$$

$$Pic = Pic + GV \ x \ G \hspace{3.2cm} \text{(Equation 12.2)}$$

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

The second node 7 authenticating the first node 3
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 (Pic) that was determined at step 430.

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 (Vic) should only be correctly verified with the corresponding first node second public key (Pic), since Vic and Pic 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

-28-

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.

The above authentication may be suitable for scenarios where one of the two nodes is a trusted node and only one of the nodes need to be authenticated. For example, the first node 3 may be a client and the second node 7 may be a server trusted by the client such as a wallet provider. Thus the server (second node 7) may need to authenticate the credentials of the client (first node 3) in order to allow the client access to the server system. It may not be necessary for the server to be authenticate the credentials of the server to the client. However in some scenarios, it may be desirable for both nodes to be authenticated to each other, such as in a peer-to-peer scenario.

The second node 7 determining the common secret

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:

$$V_{2S} = Vis + GV \qquad\qquad \text{(Equation 13.1)}$$
$$V_{2S} = Vis + \text{SHA-256}(M) \qquad\qquad \text{(Equation 13.2)}$$

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:

$$S = V_2s \times P_2c \qquad\qquad \text{(Equation 14)}$$

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

-29-

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.

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

$$S = V_2c \times P2S \qquad \text{(Equation 11)}$$
$$S = V_2c \times (V_{2S} \times G)$$
$$S = (V_2c \times V_{2S}) \times G \qquad \text{(Equation 15)}$$

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

$$S = V_2s \times P_2c \qquad \text{(Equation 14)}$$
$$S = V_2s \times (V_2c \times G)$$
$$S = (V_2s \times V_2c) \times G \qquad \text{(Equation 16)}$$

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

$$S = (V_{2C} \times V_{2S}) \times G = (V_{2S} \times V_{2C}) \times G \qquad \text{(Equation 17)}$$

The common secret (CS) and secret key

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. This communication may be used to convey part of a private key, a representation of or identifier for a private key, or mnemonic for a private key. Therefore, once the invention has been used during set-up of, for example, a digital wallet or other controlled resource, secure communication between the parties can be performed thereafter.

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 AES_{256} encryption. It could also be converted into a 160-bit integer using RIPEMD160 for any applications requiring this length key.

-30-

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. 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).

It should be noted that the above-mentioned embodiments illustrate rather than limit the invention, and that those skilled in the art will be capable of designing many alternative embodiments without departing from the scope of the invention as defined by the appended claims. In the claims, any reference signs placed in parentheses shall not be construed as limiting the claims. The word "comprising" and "comprises", and the like, does not exclude the presence of elements or steps other than those listed in any claim or the specification as a whole. In the present specification, "comprises" means "includes or consists of and "comprising" means "including or consisting of.   The singular reference of an element does not exclude the plural reference of such elements and vice-versa. The invention may be implemented by means of hardware comprising several distinct elements, and by means of a suitably programmed computer. In a device claim enumerating several means, several of these means may be embodied by one and the same item of hardware. The mere fact that certain measures are recited in mutually different dependent claims does not indicate that a combination of these measures cannot be used to advantage.

-31-

CLAIMS:

1. A computer-implemented method of controlling access to a resource, the method comprising the steps:

   splitting a verification element into a plurality of shares;

   determining a common secret at two or more nodes in a network;

   using the common secret to transmit at least one share of the verification element between the two or more nodes.

2. A method according to claim 1 wherein the verification element is a cryptographic key, a representation of a cryptographic key, or some element which may be used to access, calculate or retrieve the cryptographic key.

3. A method according to claim 1 or 2 and further comprising the step of using the common secret to generate an encryption key, preferably wherein the encryption key is used to encrypt the at least one share of the verification element or a message comprising or relating to said at least one share.

4. A method according to any preceding claim and further comprising the step of:

   storing at least three shares of the verification element at different locations relative to each other;

   preferably wherein at least one of the three shares is stored in or on a back-up or safe-storage facility which is separate, independent and/or distinct from the at least two other locations.

5. A method according to any preceding claim wherein the resource is a digital wallet or other resource relating to some form of currency.

6. A method according to any preceding claim wherein the verification element is split into a plurality of shares such that the verification element can be restored or regenerated from two or more of the shares;

-32-

preferably wherein the verification element is split into a plurality of shares using Shamir's Secret Sharing Scheme.

7. A method according to any preceding claim wherein the common secret is:
    i) determined at the at least two nodes independently of each other, such that the common secret does not require transmission over a communications channel between the nodes; and/or
    ii) shared only by the at least two nodes.

8. A method according to any preceding claim and comprising the step of setting up, creating or registering a digital wallet, wherein the verification element is associated with the digital wallet.

9. A method according to any preceding claim wherein the common secret is determined by first (C) and second (S) nodes in the network, wherein the first node (C) is associated with a first asymmetric cryptography pair having a first node master private key (Vic) and a first node master public key (Pic), and the second node (S) is associated with a second asymmetric cryptography pair having a second node master private key (Vis) and a second node master public key (Pis);

wherein the method further comprising the steps:

- determining a first node second private key (V_2c) based on at least the first node master private key (Vic) and a Generator Value (GV);
- determining a second node second public key (P_2s) based on at least the second node master public key (Pis) and the Generator Value (GV); and
- determining the common secret (CS) based on the first node second private key (V_2c) and the second node second public key (P_2s),

wherein the second node (S) has the same common secret (S) based on a first node second public key (P_2c) and a second node second private key (V_2s), wherein:

- the first node second public key (P_2c) is based on at least the first node master public key (Pic) and the Generator Value (GV); and

-33-

- the second node second private key (V_2s) is based on at least the second node
master private key (Vis) and the Generator Value (GV).

10. A method according to claim 9 wherein the Generator Value (GV) is based on a
message (M).

11. A method according to claim 10 and further comprising the steps of
- generating a first signed message (SMI) based on the message (M) and the first
node second private key (V_2c); and
- sending, over the communications network, the first signed message (SMI) to the
second node (S),
wherein the first signed message (SMI) can be validated with a first node second
public key (P_2c) to authenticate the first node (C).

12. A method according to either claim 10 or 11 and further comprising:
- receiving, over the communications network, a second signed message (SM2) from
the second node (S);
- validating the second signed message (SM2) with the second node second public key
(P2S); and
- authenticating the second node (S) based on the result of validating the second signed
message (SM2),
wherein the second signed message (SM2) was generated based on the message (M),
or a second message (M2), and the second node second private key (V_2s).

13. A method according to any one of claims 9 to 12 and further comprising:
- generating a message (M); and
- sending, over a communications network, the message (M) to the second node (S).

14. A method according to any of claims 10 to 13 further comprising:
- receiving the message (M), over the communications network, from the second
node (S).

-34-

15. A method according to any of claims 10 to 14 and further comprising:
    - receiving the message (M), over the communications network, from another node.

16. A method according to any of claims 9 to 15 wherein the first node master public key (Pic), second node master public key (Pis) are based on elliptic curve point multiplication of respective first node master private key (Vic) and second node master private key (Vis) and a base point (G).

17. A method according to any of claims 9 to 16 and further comprising the steps of:
    - receiving, over the communications network, the second node master public key (Pis); and
    - storing, at a data store associated with the first node (C), the second node master public key (Pis).

18. A method according to any of claims 9 to 17 and further comprising the steps of:
    - generating, at a first node (C), the first node master private key (Vic) and the first node master public key (Pic);
    - sending, over the communications network, the first node master public key (Pic) to the second node (S) and/or other node; and
    - storing, in a first data store associated with the first node (C), the first node master private key (Vic).

19. A method according to any of claims 9 to 18 wherein the Generator Value (GV) is based on determining a hash of a previous Generator Value (GV).

20. A method according to any of claims 9 to 19 wherein the first asymmetric cryptography pair and the second asymmetric cryptography pair are based on a function of respective previous first asymmetric cryptography pair and previous second asymmetric cryptography pair.

21. A computer-based system arranged to perform the steps of any preceding claim.

-35-

22. A system according to claim 21, wherein:

the resource is a digital wallet or other resource relating to some form of currency.

23.   A system according to claim 20 to 22, wherein the system comprises software arranged to enable the set up, creation or registration of a digital wallet, wherein the verification element is associated with the digital wallet.

24. A computer-implemented system arranged to control access to a digital wallet, the system being operative to:

determine a first entity second private key based on at least a first entity master private key and a generator value;

determine a second entity second private key based on at least a second entity master private key and the generator value;

determine a common secret (CS) at the first entity based on the first entity second private key and the second entity second public key, and determining the common secret (CS) at the second entity based on the second entity second private key and first entity second public key; and

wherein:

the first entity second public key and the second entity second public key are respectively based on at least the first/second entity master key and the generator value.
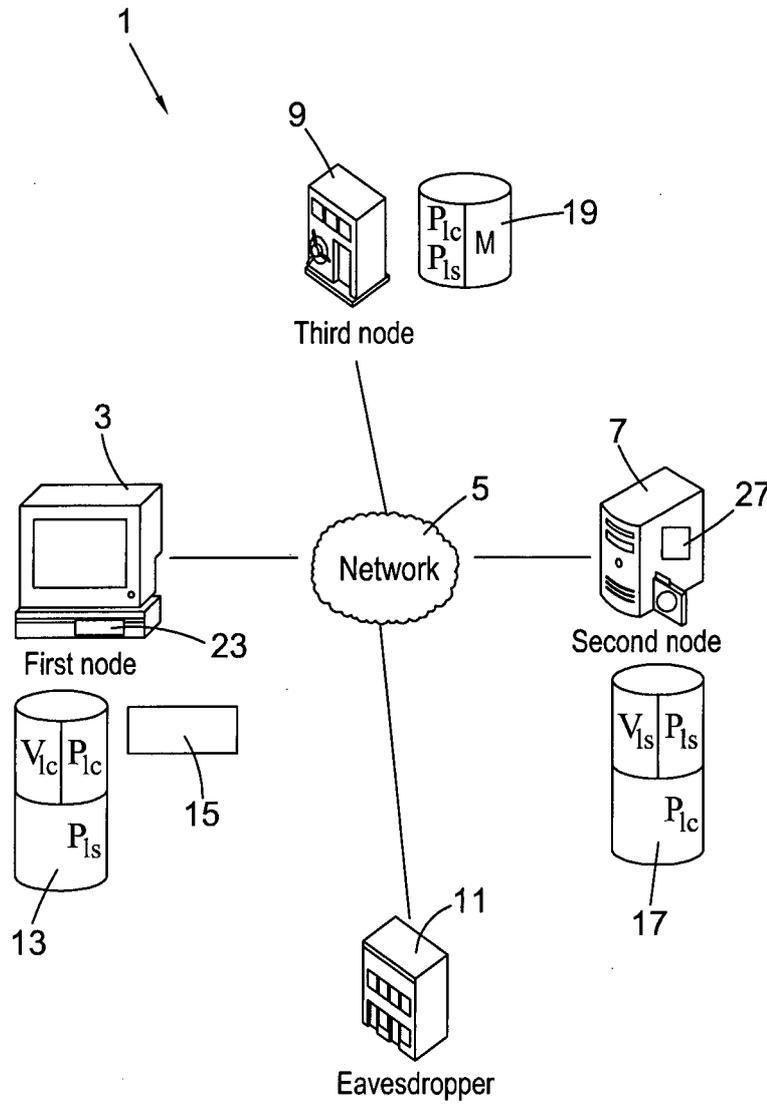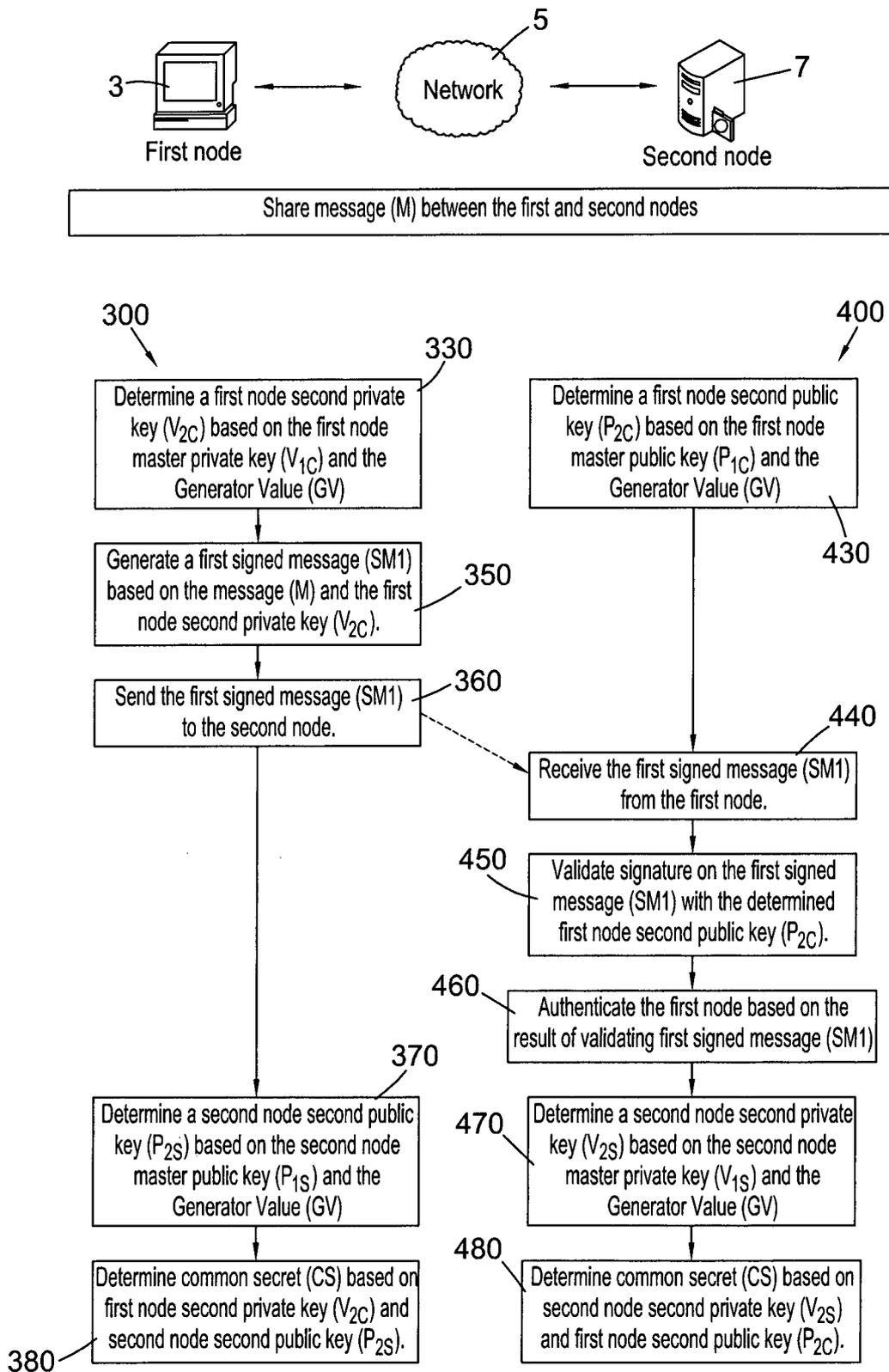
25. A method of controlling access to a digital wallet, the method comprising the steps:

determining a first entity second private key based on at least a first entity master private key and a generator value;

determining a second entity second private key based on at least a second entity master private key and the generator value;

determining a common secret (CS) at the first entity based on the first entity second private key and the second entity second public key, and determining the common secret (CS) at the second entity based on the second entity second private key and first entity second public key; and

wherein:

-36-

the first entity second public key and the second entity second public key are
    respectively based on at least the first/second entity master key and the generator
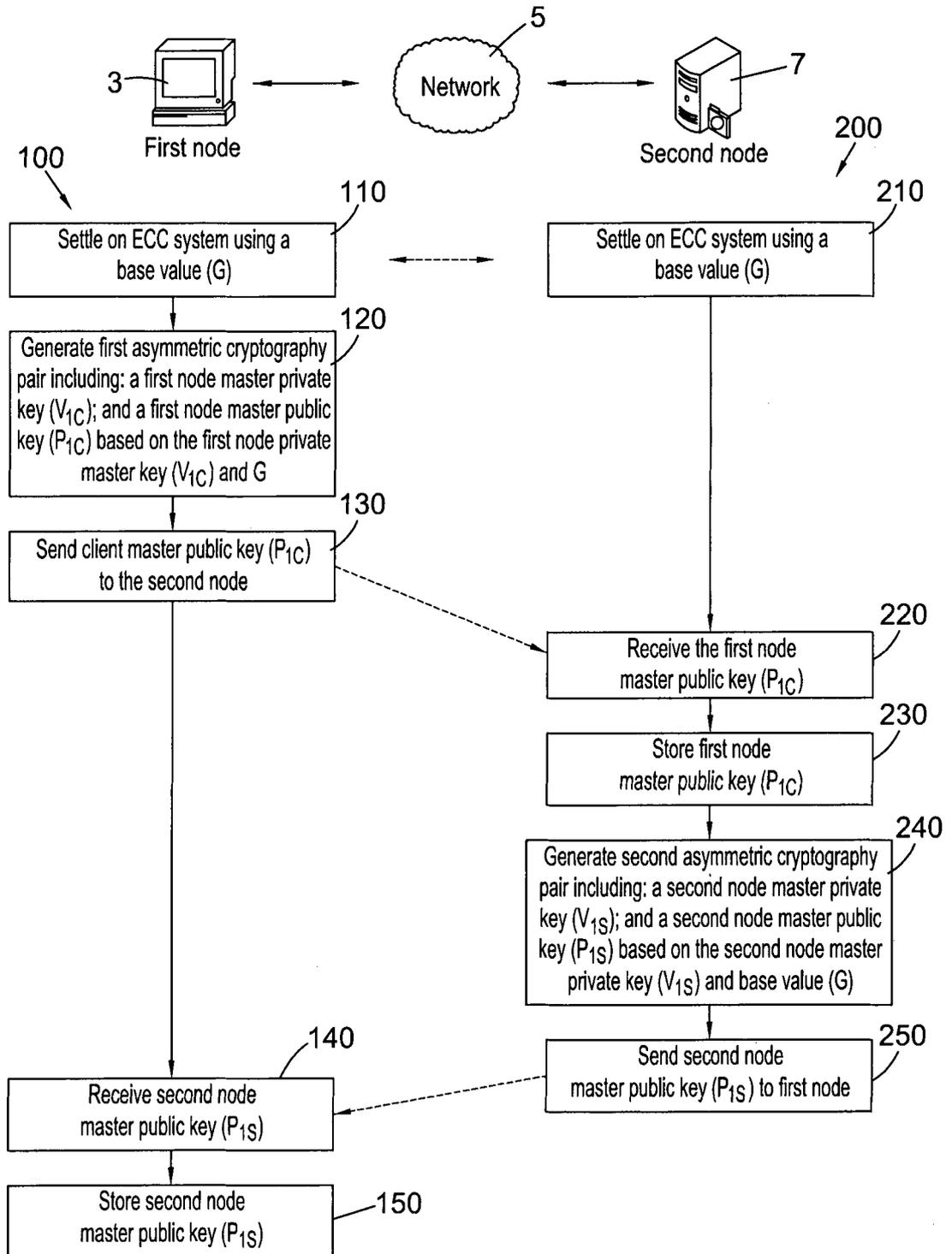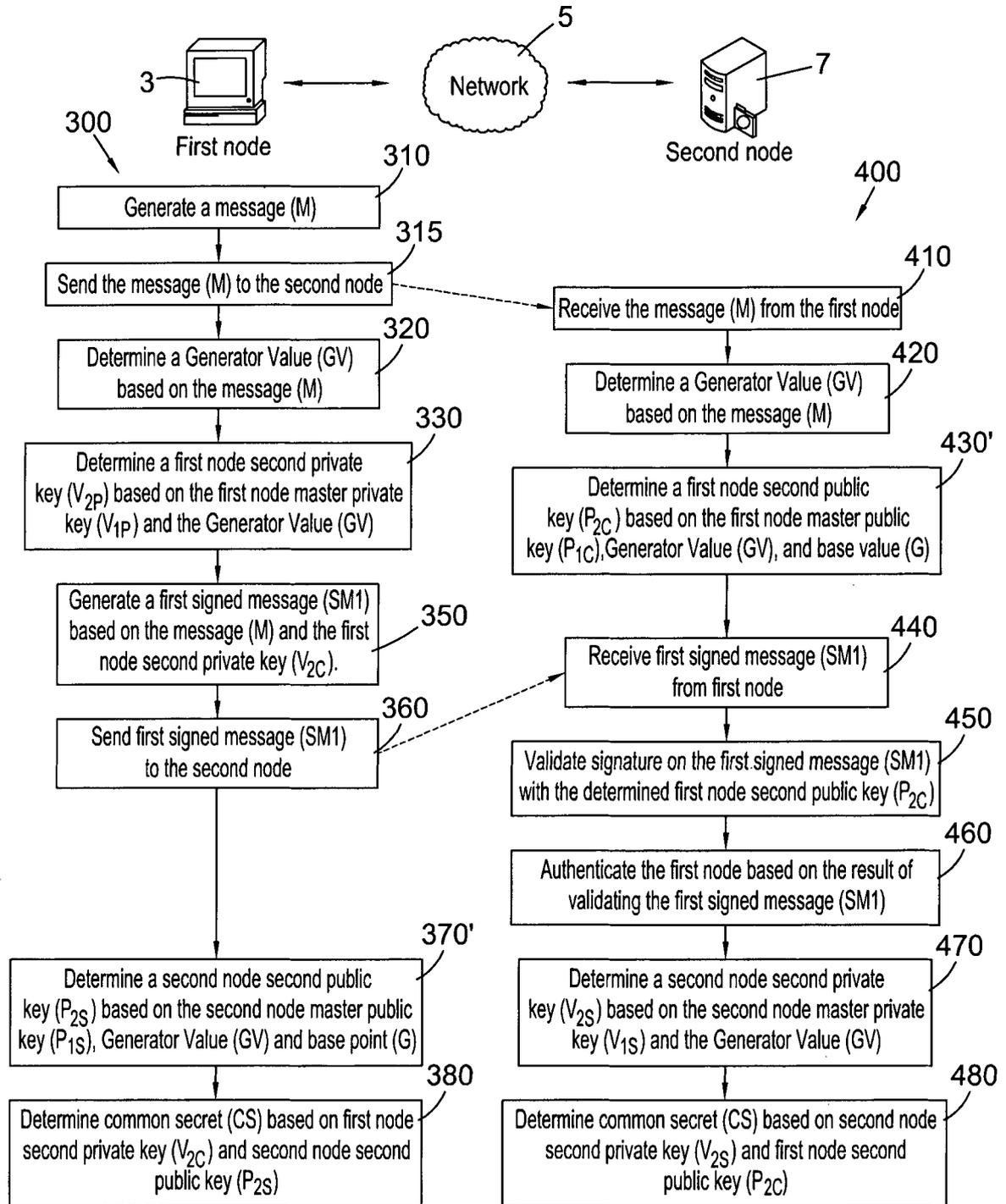   value.

5

Fig. 1

2/5

Fig. 2

3/5

Fig. 3

Fig. 4

5/5

Fig. 5

INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06Q20/38
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 G06Q

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: "Decentralizing Privacy: Using Blockchain to Protect Personal Data", THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC. (IEEE) CONFERENCE PROCEEDINGS, 1 May 2015 (2015-05-01) , page 180, XP055360065 , Piscataway Section A. From Storage to Processing ----- -/- ·	1-25

[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) or which 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
3 April 2017	12/04/2017

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

 Frank, Mario |

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.
X	Joseph Poon ET AL:; "The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments" , , 14 January 2016 (2016-01-14) , XP055358006, Retrieved from the Internet: URL: https ://lightning. network/ lightning-ne twork-paper.pdf [retrieved on 2017-03-23]	24,25
A	Sections 2, 3, 3.3, 5 -----	1-23
T	Wuille Peter: "Hierarchical Deterministic Wallets" , , 12 February 2016 (2016-02-12) , XP055358026, Retrieved from the Internet: URL: https ://github. eom/bitcoin/bips/blob/a b90b5289f0356282397fa9b8aa47d2238a7b380/bi p-0032 .mediawiki [retrieved on 2017-03-23] -----	
X	Vitalik Buterin: "Secret Sharing DAOs: The Other Crypto 2.0 - Ethereum Blog" , Ethereum Blog, 26 December 2014 (2014-12-26) , XP055360054, Retrieved from the Internet: URL: https ://blog.ethereum.Org/2014/12/26/s ecret-sharing-daos-crypto-2-0/ [retrieved on 2017-03-29] page 1 Section "Applications" -----	1-25