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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
13 December 2018 (13.12.2018)

WIPO | PCT

(10) International Publication Number
WO 2018/225053 A1

(51) International Patent Classification:
H04L 9/30 (2006.01) *H04L 9/32* (2006.01)

(21) International Application Number:
PCT/IL2018/050505

(22) International Filing Date:
09 May 2018 (09.05.2018)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/516,113 07 June 2017 (07.06.2017) US
15/727,768 09 October 2017 (09.10.2017) US

(71) Applicant: **BAR-ILAN UNIVERSITY** [IL/IL]; Ramat Gan 5290002, 5290002 Ramat Gan (IL).

(72) Inventor: **LINDELL, Yehuda**; 13 Menachem Begin St., 5442111 Givat Shmue (IL).

(74) Agent: **EREZ, Johnathan**; 3 Shalem St., 5221550 Ramat-Gan (IL).

(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, JO, 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))*

(54) Title: DIGITAL SIGNING BY UTILIZING MULTIPLE DISTINCT SIGNING KEYS, DISTRIBUTED BETWEEN TWO PARTIES

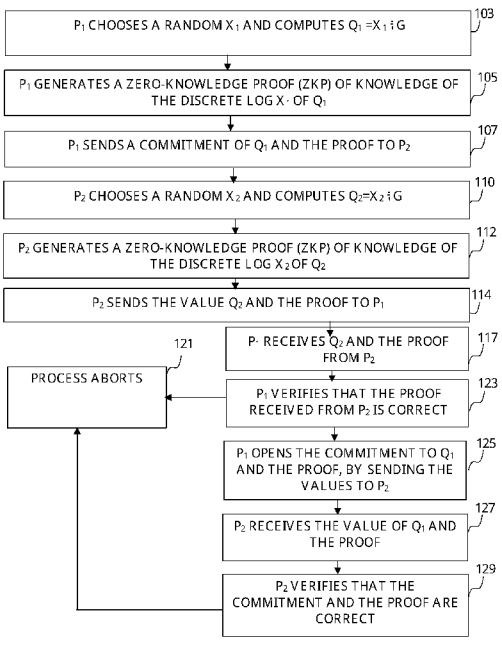

Fig. 1A

(57) Abstract: Described herein is a method and system for digital signing by utilizing Elliptic Curve Digital Signature Algorithm (ECDSA) with a group generator of an elliptic-curve group of order q, and an elliptic curve point Q. The method may be configured to receive a digital message and associated with a request from a third-party in order to sign the digital message. The system designed to sign such messages may comprise two parties denoted P_1 and P_2 configured to conduct a multiparty signing procedure by utilizing ECDSA. The digital signing procedure may follow preliminary steps configured to set the system with the necessary conditions for the multiparty signing procedure. Such preliminary steps may set the parties P_1, and P_2, in accordance with the configuration defined herein.

WO 2018/225053 A1

DIGITAL SIGNING BY UTILIZING MULTIPLE DISTINCT SIGNING KEYS, DISTRIBUTED BETWEEN TWO PARTIES

FIELD OF THE INVENTION

The invention generally relates to the field of cryptography and digital signing processes, more specifically to the usage of Elliptic Curve Digital Signature Algorithm for multiparty digital signing.

BACKGROUND OF THE INVENTION

Threshold cryptography can be used where multiple signatures are needed to generate a signature, and likewise where highly confidential documents should only be decrypted and viewed by a quorum. Furthermore, threshold cryptography can be used to provide a high level of key protection. This is achieved by sharing the key on multiple devices (or between multiple users) and carrying out private-key operations via a secure protocol that reveals nothing but the output. This provides key protection since an adversary needs to breach multiple devices in order to obtain the key. Threshold cryptography is of practical use, as can be seen by the fact that a number of startup companies are now deploying threshold cryptography for the purpose of key protection. One example use is due to the fact that Elliptic Curve Digital Signature Algorithm (ECDSA) signing is used in Bitcoin, and the theft of a signing key can be immediately translated into concrete financial loss. Bitcoin has a multi-signature solution built in, which is based on using multiple distinct signing keys rather than a threshold signing scheme. Nevertheless, a more general solution is obtained via threshold cryptography. Fast threshold cryptography protocols exist for a wide variety of problems, including RSA signing and decryption, ElGamal and ECIES encryption, Schnorr signatures, Cramer-Shoup, and more. Despite being a widely-used standard, DSA/ECDSA with distributed key shares has resisted attempts at constructing efficient protocols for threshold signing. This is due to the difficulties to calculate the curve points x_1 and y_1 without knowing the parameter k utilized to calculate the points with G, the generator of the elliptic curve.

SUMMARY OF THE INVENTION

The present invention discloses a method and system for digital signing by utilizing ECDSA with a group generator of an elliptic-curve group of order q, and an elliptic curve point Q. The

method disclosed in the present invention can be configure to receive a digital message and associated with a request from a third-party in order to sign the digital message. The system designed to sign such messages may comprise two parties denoted P_1 and P_2 configured to conduct a multiparty signing procedure by utilizing ECDSA. The digital signing procedure may follow some preliminary steps configured to set the system with the necessary conditions for the multiparty signing procedure. Such preliminary steps may set the parties P_1, and P_2, with the following configuration:

- P_1 may hold a first random share set to compute a first point on said elliptic curve, an elliptic curve point Q on the elliptic curve, a public key and a private key of an additively homomorphic encryption scheme, and a group generator of an elliptic-curve group of order q to compute elliptic curve points.
- P2 may hold: a second random share, set to generate a second point on said elliptic curve, said elliptic curve point Q on said elliptic curve, said public key, and a ciphertext created by encrypting said first random share with said public key, group generator of an elliptic-curve group of order q to compute elliptic curve points.

The parties P_1 and P_2 may also be configured to perform the preliminary steps configured to set the system wherein, the first random share was not exposed to P_2 in a nonencrypted format during the signing process, and the second random share was not exposed to P_1 in a nonencrypted format during the signing process. In some case, P_1 and P_2 may also be configured to creating a hash value of the digital message with one hash function by P_1, and creating a hash value of the digital message with said one hash function by P_2.

According to possible embodiments of the present invention, once the message and the request are received from a third party, the parties P1 and P2 which may be configured according to the preliminary steps, may conduct the digital signing procedure by performing the following: P_1 can choose a random value K_1, and utilize the group generator and the random value K_1 to compute an elliptic curve point R_1 on the elliptic. Then P_1 can send the elliptic curve point R_1 to P_2. Similarly, P_2 can choose a random value K_2 and utilizes the group generator and the random value K_2 to compute an elliptic curve point R_2 on the elliptic. Then, P_2 can utilize R_1 and K_2 to compute an elliptic curve point R on said elliptic curve. Then, P_2 can choose a random value and utilize an additive homomorphic scheme with the public key to computes a value C which may

comprise: the hash value of the digital message, first random share, second random share, k_1, k_2 and the random value, wherein value C is encrypted with said public key. After the encryption of the value C, P_2 can send the encrypted value C to P_1. Then, P_1 can decrypt the encrypted value C with the private key, and then, utilize the content in value C, the elliptic curve point R_1, and the random value K_1 to produce a digital signature of the digital message. Once the digital signature of the digital message is computed, P_1 can send it to the third-party who requested the signing procedure.

The system disclosed in the present invention may comprises two parties configured to receive requests for digital signing procedures. The requests for digital signing may arrive to the system from third-party entities, such as servers operating overt the internet, computerize device configured to provide such requests, and the like. In some cases, the parties P_1 and P_2 may comprise software application for managing the cryptographic processes in the system. In some embodiments of the present invention the parties P_1 and P_2 may be configured with a key storage media designed to store keys, secrets utilized in the security process, key shares, and the like.

BRIEF DESCRIPTION OF THE DRAWINGS

Some embodiments of the invention are herein described, by way of example only, with reference to the accompanying drawings. With specific reference now to the drawings in detail, it is stressed that the particulars shown are by way of example and for purposes of illustrative discussion of embodiments of the invention. In this regard, the description taken with the drawings makes apparent to those skilled in the art how embodiments of the invention may be practiced.

In the drawings:

Fig. 1A discloses a method denoted KeyGen_1 for choosing random shares between two parties P_1 and P_2 and verifying that P_1 and P_2 behave honestly, according to exemplary embodiments of the present invention;

Fig. 1B discloses a method denoted KeyGen_2 for generating a distributed key between two parties P_1 and P_2 under the assumption that one of P_1 and P_2 may behave dishonestly, using the random shares generated in KeyGen_1, by using an additively homomorphic encryption scheme, according to the exemplary embodiments of the present invention;

Fig. 2 discloses a procedure denoted Sign(M) for digital signing by utilizing a key distributed between two parties P_1 and P_2, under the assumption that one of P_1 and P_2 may behave dishonestly, according to exemplary embodiments of the present invention;

Fig. 3 discloses a method for multiparty digital signing by utilizing ECDSA based on using a signing key distributed between two parties according to exemplary embodiments of the present invention, and;

Fig. 4 discloses a system designed to perform a multiparty digital signing by utilizing ECDSA based on using a signing key distributed between two parties, according to exemplary embodiments of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

The present invention discloses a method and system for multiparty digital signing by utilizing Elliptic Curve Digital Signature Algorithm (ECDSA) based on signing keys, distributed between two parties, denoted as P_1 and P_2. The method and system disclosed in the present invention may be configured to conduct a multiparty signing procedure with preliminary steps comprising: a step for choosing random shares between two parties P_1 and P_2 ensuring that they behave honestly, and a step for generating the distributed signing keys between P_1 and P_2. The preliminary steps may be conducted by the parties P_1 and P_2, wherein said parties utilize a zero-knowledge proof of a knowledge of the shares the parties hold. The parties P_1 and P_2 can be configured to generate an elliptic curve point Q such that $Q = x \cdot G$, wherein G is the generator point of an Elliptic-curve group of order q. In some cases, the computations performed by P1 and P2 in the method disclosed at the present invention may utilize operators that are homomorphic properties provided by the encryption scheme, which are known to a person who has ordinary skills in the art.

Fig. 1A discloses a method denoted KeyGen_1 for choosing random shares between two parties P_1 and P_2 and verifying that P_1 and P_2 behave honestly, according to exemplary embodiments of the present invention. In step 103 P_1 chooses a random $x_1 \leftarrow Z_q$ and computes $Q_1 = x_1 \cdot G$, where G is the generator point of an Elliptic-curve group of order q, and Z_q is defined as the multiplicative group of integers modulo q. In step 105 P_1 generates a zero-knowledge proof (denoted as ZKP) of knowledge of the Discrete Log of Q_1 which is equal to x_1. In some embodiments of the present invention, the parties P1 and P2 may utilize a non-interactive zero-knowledge proof which requires a proof generation performed by one party such as P_1, or P_2, and

verification performed by the other party. In some other cases, the parties may utilize an interactive zero-knowledge proof. In some cases, this zero-knowledge proof may be the Schnorr protocol, known to a person who has ordinary skills in the art. In step 107 P_1 sends P_2 a commitment of Q_1 and the proof. The proof may be the proof resultant in step 105 above. In some cases, the commitment sent by P_1 may be a non-interactive commitment given by one party to the other. For example, such a commitment can be the hash value resultant on Q_1 and the proof, and a random string generated or received by P_1. Such a commitment is opened by P_1 sending Q_1 and the proof and the random string to P_2, and P_2 can verify that the commitment was correct by re-computing the hash and comparing to the commitment hash value. In step 110 P_2 chooses a random $x_2 \leftarrow Z_q$ and computes $Q_2=x_2 \cdot G$, where G is the generator point of an Elliptic-curve group of order q. In step 112 P_2 generates a ZKP of knowledge of the Discrete Log of Q_2 which is equal to x_2. In step 114 P_2 sends to P_1 the value Q_2 and the proof resultant in step 112 above. In step 117 P_1 receives the proof from P_2. In step 123 P_1 verifies that the proof received from P_2 is correct using the verification procedure specified for the proof. Such verification procedures can be defined by a person who has ordinary skills in the art. In case the proof is not correct, P_1 performs step 121 and aborts. In other cases, P_1 can continue to step 125 of the method and open the commitment to Q_1 and the proof. If the commitment is computed as described above as a hash value, then it is opened by sending Q_1 and the proof and the random string to P_2. In step 127 P_2 receives the value Q_1 and the proof. In step 129 P_2 verifies that the proof of the ZKP associated with Q_1 is correct, and that the commitment is correct. Thus, in case either the proof or the commitment are not corrected, P_2 performs the step 121 of the method and aborts.

Fig. 1B discloses a method denoted KeyGen_2 for generating a distributed key between two honest parties P_1 and P_2, using the random shares generated in KeyGen_1, by using an additively homomorphic encryption scheme, according to the exemplary embodiments of the present invention. In step 131 P_1 holds a share x_1 in Z_q where $Q_1=x_1 \cdot G$ and G is the generator point of an Elliptic-curve group Q of order q. P_2 also holds a share x_2 in Z_q where $Q_2=x_2 \cdot G$ and G is the generator point of the same Elliptic-curve group Q of order q. In step 135 P_1 generates a public key denoted pk and a private key denoted sk for the public key of an additively homomorphic encryption scheme. The homomorphic encryption scheme may support addition of ciphertexts and multiplication by a scalar. The additively homomorphic encryption scheme can be used by

any implementation of additively homomorphic encryption scheme known to a person who has ordinary skills in the art. Such an exemplary implementation may be an implementation of lattice-based schemes, such as Regev's implementation of lattice-based schemes. In some cases, the Paillier encryption scheme can be used for the same purpose.

Once P_1 holds the public key pk, a parameter denoted C_{key} can be computed such that $C_{key} = Enc_{pk}(x_1)$, wherein $Enc_{pk}()$ is the encryption process utilizing the public key pk. In step 140 P_1 generates a ZKP to prove that N, defined as N=pk is a valid RSA modulus which is also a proof that the public key pk was correctly generated. In step 145 P_1 generates a ZKP of knowledge to prove that C_{key} is an encryption of the unique x_1 such that $Q_1 = x_1 \cdot G$. Such zero-knowledge proofs can be designed using standard techniques known to any person who has ordinary skills in the art. In step 150 P_1 sends the proof resultant of the process in step 140 which proved that N is a valid RSA modulus, and the proof resultant of the process in step 145 which proved that C_{key} is an encryption of the unique x_1 such that $Q_1 = x_1 \cdot G$, to P_2. In step 153 P_2 receives the proofs sent at step 153. In step 155 P2 verifies the proofs received in step 153 according to the specified verification technique of the proof. In case the verification fails, P_2 aborts. In step 157 P_2 verifies that the public key pk = N is of length at least min (3log|q|, n), where q is the order of the Elliptic curve group, and n denotes the standard recommended length of RSA public keys, and in case the verification fails, the process aborts. In step 160 P_1 computes $Q = x_1 \cdot Q_2$ and stores (x_1, Q) and stores (x_1, Q, pk, sk), and in step 165 P_2 computes $Q = x_2 \cdot Q_1$ and stores (x_1, Q, C_{key}.),

Fig. 2 discloses a procedure denoted Sign(M) for digital signing by utilizing a key distributed between two parties, P_1 and P_2, under the assumption that one of P_1 and P_2 may behave dishonestly, according to exemplary embodiments of the present invention. In step 205 a message for signing is prepared by a computerized application. Such an application may be operated by one of the parties, which may require to sign a digital message or document. In such a case, P_1 and P_2 compute: $M' \leftarrow H_q(M)$. In some cases, the $H_q()$ may be a hash function known to both parties P_1 and P_2, where H_q maps strings of arbitrary length to Z_q.

In step 210 P_1 chooses a random k_1, computes $R_1 = k_1 \cdot G$ and then computes a ZKP of knowledge of the Discrete Log k_1 of R_1. In step 215 P_1 sends a commitment of R_1 and the proof to P_2. Such a commitment may be the hash of the value, the proof and a random string, as described above. In step 220 P_2 chooses a random k_2, computes $R_2 = k_2 \cdot G$ and then computes a

ZKP of knowledge of the Discrete Log k_2 of R_2. In step 225 P_2 sends the value R_2 and the proof, which may be the Discrete Log k_2 of R_2, to P_1. In step 230 P_1 receives R_2 and the proof from P_2 and verifies that the Discrete Log of R_2 is correct using the specified verification of the proof. In case the proof is not correct, P_1 aborts the method. In step 235 P_1 opens the commitment to R_1 and the proof. For example, if the commitment uses a hash, then P_1 sends R_1, the proof and the random string. In step 240 P_2 receives R_1 and the proof, verifies the commitment (by re-computing the hash and comparing) and validates that the proof is correct using the proof verification specification. In step 250 P_2 computes $R = k_2 \cdot R_1$ and $r = r_x \bmod q$, wherein $R = (r_x, r_y)$ by the Elliptic curve point definition, and q is the order of the Elliptic curve group. In step 255 P_2 chooses a random $\rho \leftarrow Z_{q2}$ and uses the additively homomorphic properties (such as those provided by Paillier encryption or any other additively homomorphic encryption schemes) to compute an encryption defined by $Enc_{pk}(\rho \cdot q + [(k_2)^{-1} \cdot M' + (k_2)^{-1} \cdot R \cdot x_2 \cdot x_1 \bmod q])$. This is computed using C_{key} in the following steps: (1) compute $C_1 \leftarrow Enc_{pk}(\rho \cdot q + [(k_2)^{-1} M' \bmod q])$ using the additively homomorphic encryption scheme, (2) compute $v \leftarrow (k_2)^{-1} \cdot r \cdot x_2 \bmod q$, and $C_2 \leftarrow v \odot C_{key}$ using the homomorphic property of additive encryption that enables multiplication by a scalar, (3) computes $C = C_1 \oplus C_2$ using the homomorphic property of additive encryption that enables the addition of two ciphertexts. In step 260 P_2 sends the value C to P_1 and then in step 265 P_1 computes $R = k_1 \cdot R_2$, $S' = Dec_{sk}(C)$ and $S = (k_1)^{-1} \cdot S' \bmod q$, wherein $Dec_{sk}()$ is the decryption function utilizes the private key sk for the additively homomorphic public key scheme. In step 270 P_1 verifies that (r, s) is a valid signature on message m, using the standard ECDSA signature verification procedure, otherwise P_1 aborts.

Fig. 3 discloses a method for multiparty digital signing by utilizing ECDSA based on using multiple distinct signing keys, distributed between two parties according to exemplary embodiments of the present invention. In step 305 a third-party entity requests to sign a digital message or a document denoted as M, by P_1 and P_2. In some cases, signing on M may require a digital signing by more than one person, or a joint signature provided by more than one signees. For example, bitcoin transactions or other transactions may require multi-signatures.

P_1 and P_2 may be computerized devices configured to digitally sign electronic messages or documents. In some cases, such a third-party entity may be a server or other computerized device configured to verify the authenticity of the digital signature associated with a digital message or

a document. In some cases, P_1 and P_2 may be may be computerized devices operated by two different people, which may need to provide multiparty digital signing.

In step 310 P_1 verifies that (x_1, Q, pk, sk) are stored according to method KeyGen_2 and message M is received. In some embodiments of the present invention a session ID, denoted as a sid, may be provided by the third-party entity. The provided sid can be an integer number functioning as a counter, a unique string sequence, and the like, which define the multiparty digital signing session. In some cases, P_1 and P_2 may utilize the sid inside the zero-knowledge proofs to ensure that the sid is unique to the current multiparty digital signing session. In step 315 P_2 verifies (x_2, Q), pk and C_{key} are stored according to method KeyGen_2. In step 320 P_1 and P_2 perform the procedure Sign(M) and receive the ECDSA signature (r, s) on message M. In step 325 P_1 and P_2 send the signature (r, s) and the message M. In step 330 the third-entity verifies the authenticity of the signature by using a standard ECDSA signature verification procedure. In some cases, the third-party which verifies the authenticity of the signature may be configured to operate a procedure of ECDSA signature verification. For example, the third-party entity may have the public-key on the curve wherein Q is a point at the curve.

Fig. 4 discloses a system designed to perform a multiparty digital signing by utilizing ECDSA based on using one or more shared signing keys distributed between two parties, according to exemplary embodiments of the present invention. Fig. 4 shows a multiparty digital signing system 405 comprising a multiparty digital signing party 415 and a multiparty digital signing party 410. The multiparty digital signing parties 410 and 415 can be computerized devices operated singlehandedly by persons. For example, in some cases, the multiparty digital signing party 415 may be operated by one person and the multiparty digital signing party 410 may be operated by another person. The multiparty digital signing parties 415 and 410 may also have cryptographic modules 435 and 430 respectively. The cryptographic modules 435 and 430 may be configured for processing cryptographic procedures such as an encryption, security algorithms, generating secret shares, signing digital messages or digital documents and the like. The multiparty digital signing parties 415 and 410 can communicate via connection 450 which may be a telecommunication network connection such as an internet protocol based connection, a computer network based connection, and the like. The multiparty digital signing parties 415 and 410 also comprise key storages 425 and 420 respectively configured to store the key shares

and the secrets utilized in the digital signing processes. For example, in some cases, the multiparty digital signing party 415 may utilize the key storage 425 to store the (x_1, Q) required at the digital signing process, and the multiparty digital signing party 415 may store the (x_2, Q) required at the same digital signing process. The key storages 425 and 420 may be a data storage device, a magnetic storage such as hard disk, a solid-state storage, other media comprising a non-volatile memory, and the like.

In some cases, one of the parties either multiparty digital signing party 415 or multiparty digital signing party 410 may launch the KeyGen_1 and KeyGen_2 to generate the distributed keys between multiparty digital signing parties 410 and 415. In some cases, the multiparty digital signing party 410 may launch the Sign(M) procedure for signing on a message M, per request provided by the third-party entity 470. In some other cases, the multiparty digital signing party 415 may be the party which launches the Sign(M) procedure for signing on a message M, per request provided by the third-party entity 470. In such cases, the multiparty digital signing party 415 and multiparty digital signing party 410 may utilize the key storage 425 and the key storage 420, respectively to store the cryptographic data. The cryptographic data may be keys, key shares, parameters related to the elliptic-curve computation, numbers, and the likes.

The third-party entity 470 may be a computerize device seeking to receive a message M signed by a person or persons operating the multiparty digital signing parties 410 and 415. In some cases, the message M to sign may be sent by the third-party entity 470 to one of the multiparty digital signing parties 410 or 415. Then, the party received the request to sign the message M may launch the digital signing process and conduct said digital signing process with the third-party entity 470.

While the disclosure has been described with reference to exemplary embodiments, it will be understood by those skilled in the art that various changes may be made and equivalents may be substituted for elements thereof without departing from the scope of the invention. In addition, many modifications may be made to adapt a particular situation or material to the teachings without departing from the essential scope thereof. Therefore, it is intended that the disclosed subject matter not be limited to the particular embodiment disclosed as the best mode contemplated for carrying out this invention, but only by the claims that follow.

CLAIMS

1. A method for digital signing by utilizing ECDSA with a group generator of an elliptic-curve group of order q, and an elliptic curve point Q, comprising:

receiving a digital message and a request from a third-party to sign said digital message by two parties denoted P1 and P2 configured to perform digital signing by utilizing ECDSA, wherein:

(i) P1 holds a first random share set to compute a first point on said elliptic curve, a public key and a private key of an additively homomorphic encryption scheme, the elliptic curve point Q,

(ii) P2 holds a second random share, set to generate a second point on said elliptic curve, said public key, and a ciphertext created by encrypting said first random share with said public key, the elliptic curve point Q,

creating a hash value of the digital message using a hash function operated by P1, and creating a hash value of the digital message with said hash function operated by P2;

preforming a digital signing procedure by P1 and P2, wherein said digital signing procedure comprises:

using a random value K1 chosen by P1 to compute an elliptic curve point R1 on said elliptic curve,

sending elliptic curve point R1 from P1 to P2,

using a random value K2 chosen by P2 to compute an elliptic curve point R2,

utilizing elliptic curve point R2 and random value K2 by P2 to compute an elliptic curve point R,

using additive homomorphic scheme by P2 to compute a value C, wherein said value C comprising: the hash value of the digital message, first random share, second random share, random value K1, random value K2 and a random value chosen by P2, and wherein value C is encrypted with said public key,

sending value C from P2 to P1,

decrypting value C with the private key by P1, and utilizing the content in value C to produce a digital signature of the digital message;

sending the digital signature of the digital message and the hash value of the digital message to the third-party.

2. The method of claim 1, wherein the digital message and the request from the third-party also comprise a session ID.
3. The method of claim 1, wherein the first random share is not exposed to P2 in a nonencrypted format during the signing process.
4. The method of claim 1, wherein the second random share was not exposed to P1 in a nonencrypted format during the signing process.
5. The method of claim 1, wherein the hash value is created independently by P1 with the one hash function.
6. The method of claim 1, wherein the hash value is created independently by P2 with the one hash function.
7. The method of claim 1, wherein the digital signature of the digital message is sent by P1.
8. The method of claim 1, wherein the additive homomorphic scheme is based on implementation of lattice-based scheme.
9. The method of claim 1, wherein the additive homomorphic scheme is based on Paillier encryption scheme.
10. A system designed to perform digital signing by utilizing ECDSA with a group generator of an elliptic-curve group of order q, and an elliptic curve point Q, comprising:
two parties denoted P1 and P2 designed to perform the ECDSA and configured to receive requests from third-parties to sign digital messages, wherein:
    (i) P1 holds: a first random share set to compute a first point on said elliptic curve, a public key and a private key of an additively homomorphic encryption scheme,
    (ii) P1 is configured to compute a hash value of the digital message with one hash function,
    (iii) P2 holds: a second random share, set to generate a second point on said elliptic curve, said public key, and a ciphertext created by encrypting said first random share with said public key,
    (iv) P2 is configured to compute a hash value of the digital message with one hash function,
and, wherein P1 and P2 are configured to perform a digital signing procedure comparing:
P1 chooses a random value K1, and utilizes said K1 to compute an elliptic curve point R1 on said elliptic, and sends R1 to P2,

P2 chooses a random value K2 and utilizes said K1 to compute an elliptic curve point R2 on said elliptic,

P2 utilizes R1 and K2 to compute an elliptic curve point R on said elliptic curve,

P2 chooses a random value,

P2 utilizes additive homomorphic scheme with said public key and computes a value C comprising: the hash value of the digital message, first random share, second random share, k1, k2 and the random value, wherein value C is encrypted with said public key,

P2 sends the encrypted value C to P1,

P1 decrypts the encrypted value C with the private key, utilizes the content in value C, the elliptic curve point R1, and the random value K1 to produce a digital signature of the digital message;

P1 sends the digital signature of the digital message and the hash value of the digital message to the third-party.

11. The method of claim 10, wherein the additive homomorphic scheme is based on implementation of lattice-based scheme.
12. The method of claim 10, wherein the additive homomorphic scheme is based on Paillier encryption scheme.

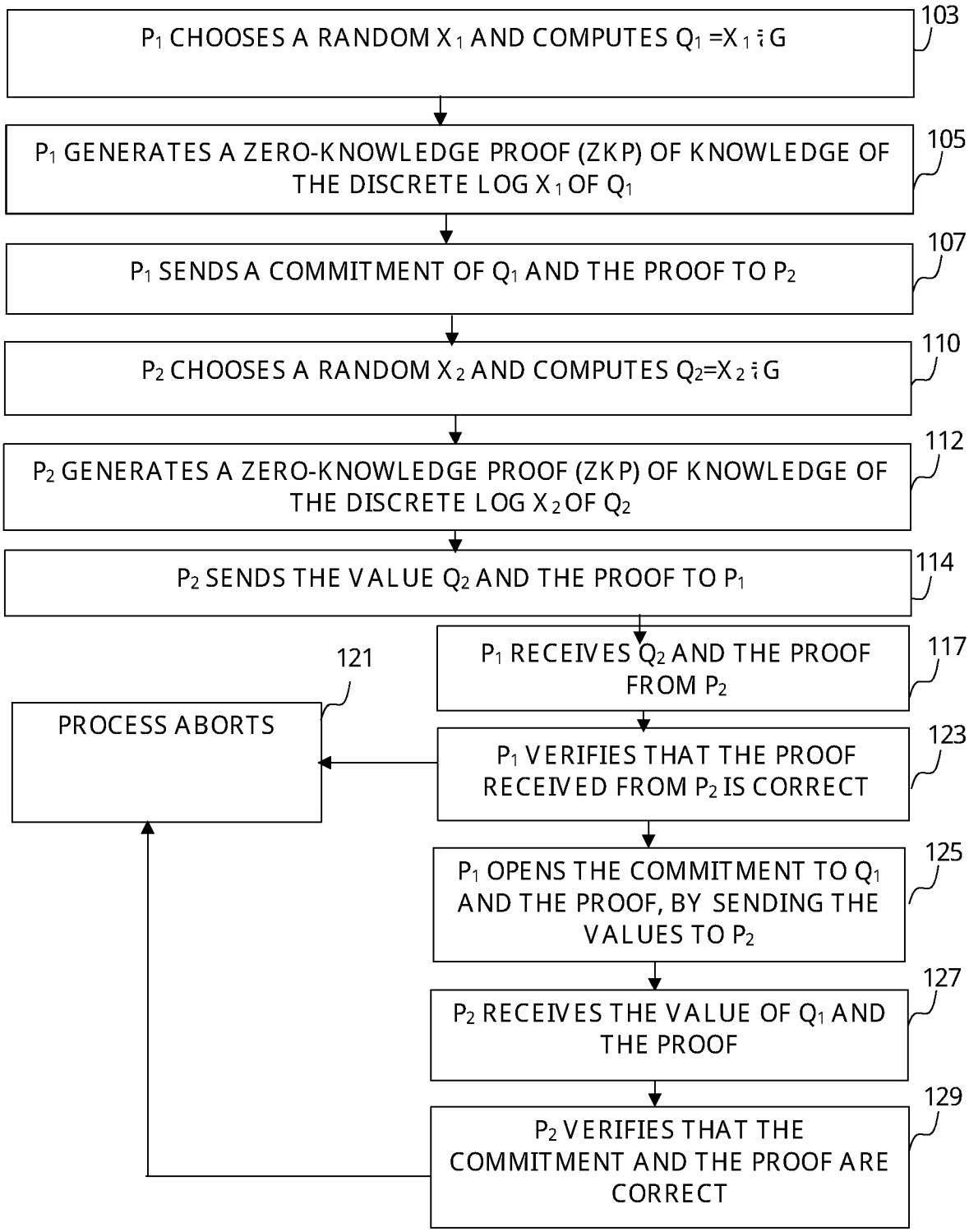
P1 CHOOSES A RANDOM X1 AND COMPUTES Q1 =X1 ·G
103
P1 GENERATES A ZERO-KNOWLEDGE PROOF (ZKP) OF KNOWLEDGE OF THE DISCRETE LOG X1 OF Q1
105
P1 SENDS A COMMITMENT OF Q1 AND THE PROOF TO P2
107
P2 CHOOSES A RANDOM X2 AND COMPUTES Q2=X2 ·G
110
P2 GENERATES A ZERO-KNOWLEDGE PROOF (ZKP) OF KNOWLEDGE OF THE DISCRETE LOG X2 OF Q2
112
P2 SENDS THE VALUE Q2 AND THE PROOF TO P1
114
P1 RECEIVES Q2 AND THE PROOF FROM P2
117
121
PROCESS ABORTS
P1 VERIFIES THAT THE PROOF RECEIVED FROM P2 IS CORRECT
123
P1 OPENS THE COMMITMENT TO Q1 AND THE PROOF, BY SENDING THE VALUES TO P2
125
P2 RECEIVES THE VALUE OF Q1 AND THE PROOF
127
P2 VERIFIES THAT THE COMMITMENT AND THE PROOF ARE CORRECT
129

Fig. 1A

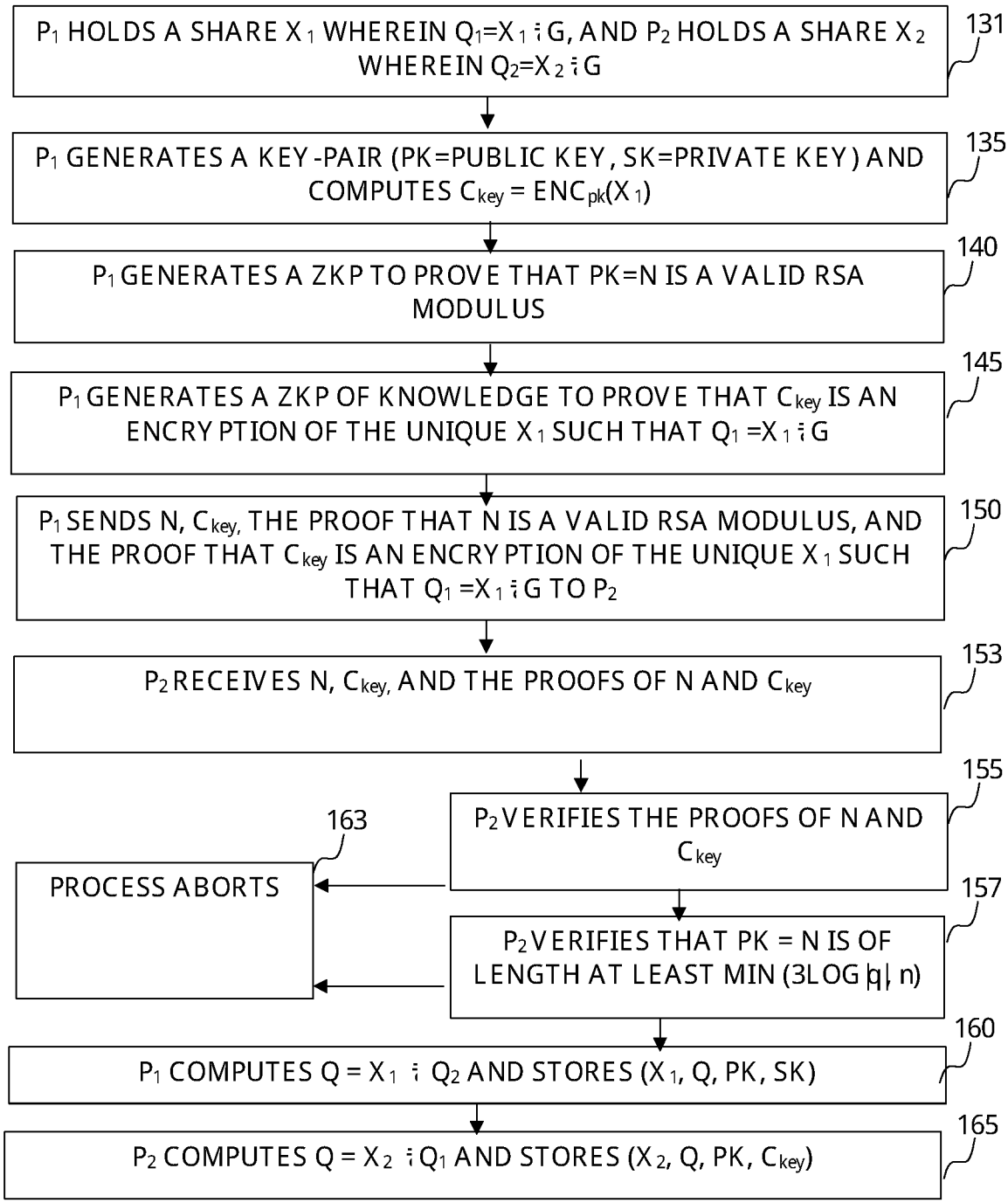
P1 HOLDS A SHARE X1 WHEREIN Q1=X1 ·G, AND P2 HOLDS A SHARE X2 WHEREIN Q2=X2 ·G
131
P1 GENERATES A KEY-PAIR (PK=PUBLIC KEY, SK=PRIVATE KEY) AND COMPUTES Ckey = ENCpk(X1)
135
P1 GENERATES A ZKP TO PROVE THAT PK=N IS A VALID RSA MODULUS
140
P1 GENERATES A ZKP OF KNOWLEDGE TO PROVE THAT Ckey IS AN ENCRYPTION OF THE UNIQUE X1 SUCH THAT Q1 =X1 ·G
145
P1 SENDS N, Ckey, THE PROOF THAT N IS A VALID RSA MODULUS, AND THE PROOF THAT Ckey IS AN ENCRYPTION OF THE UNIQUE X1 SUCH THAT Q1 =X1 ·G TO P2
150
P2 RECEIVES N, Ckey, AND THE PROOFS OF N AND Ckey
153
P2 VERIFIES THE PROOFS OF N AND Ckey
155
163
PROCESS ABORTS
P2 VERIFIES THAT PK = N IS OF LENGTH AT LEAST MIN (3LOG|q| n)
157
P1 COMPUTES Q = X1 · Q2 AND STORES (X1, Q, PK, SK)
160
P2 COMPUTES Q = X2 ·Q1 AND STORES (X2, Q, PK, Ckey)
165

Fig. 1B

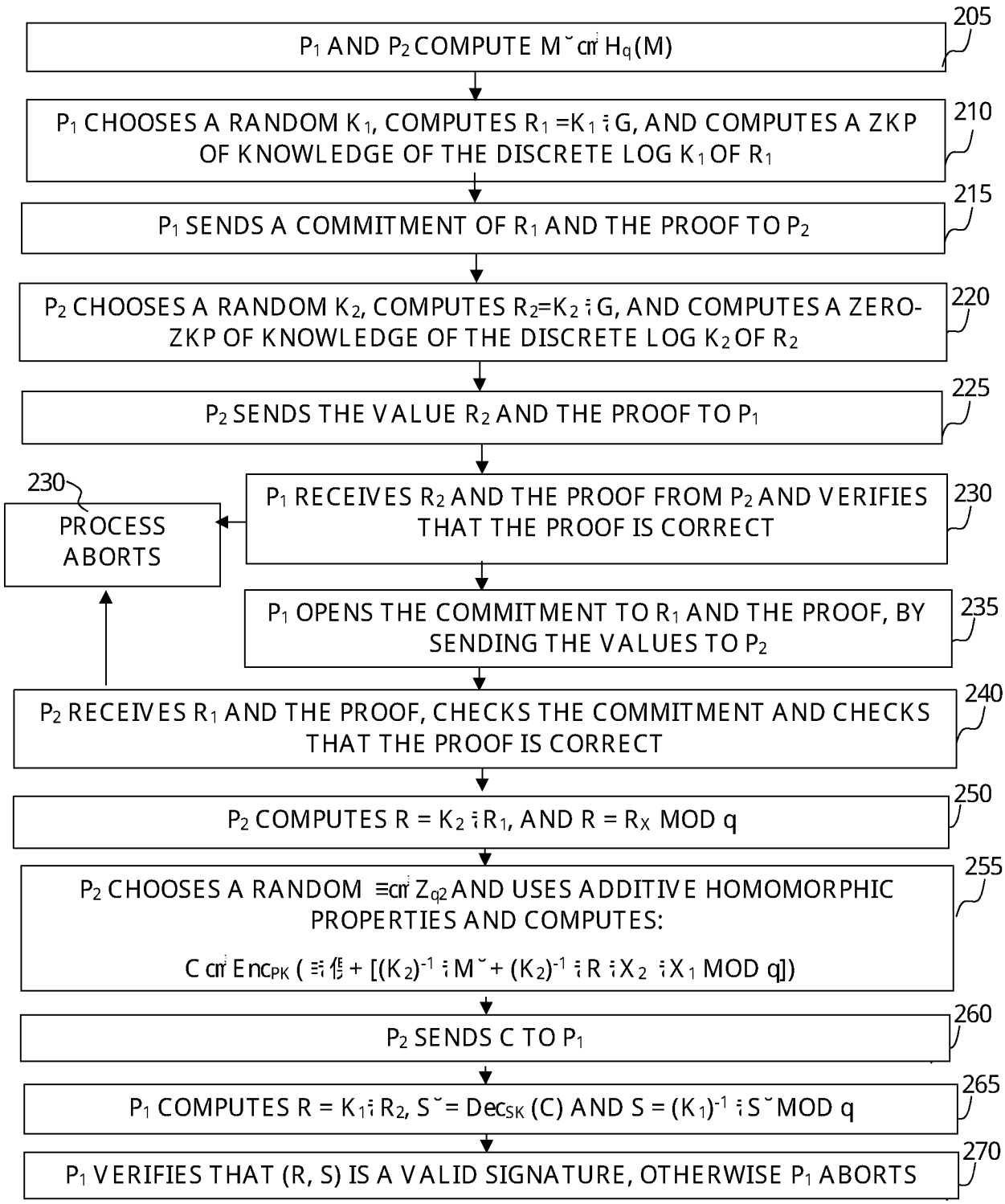
205
P1 AND P2 COMPUTE M˘cm Hq(M)
210
P1 CHOOSES A RANDOM K1, COMPUTES R1 =K1 ¡G, AND COMPUTES A ZKP OF KNOWLEDGE OF THE DISCRETE LOG K1 OF R1
215
P1 SENDS A COMMITMENT OF R1 AND THE PROOF TO P2
220
P2 CHOOSES A RANDOM K2, COMPUTES R2=K2 ¡G, AND COMPUTES A ZERO-ZKP OF KNOWLEDGE OF THE DISCRETE LOG K2 OF R2
225
P2 SENDS THE VALUE R2 AND THE PROOF TO P1
230
P1 RECEIVES R2 AND THE PROOF FROM P2 AND VERIFIES THAT THE PROOF IS CORRECT
230
PROCESS ABORTS
235
P1 OPENS THE COMMITMENT TO R1 AND THE PROOF, BY SENDING THE VALUES TO P2
240
P2 RECEIVES R1 AND THE PROOF, CHECKS THE COMMITMENT AND CHECKS THAT THE PROOF IS CORRECT
250
P2 COMPUTES R = K2 ¡R1, AND R = RX MOD q
255
P2 CHOOSES A RANDOM ≡cm Zq2 AND USES ADDITIVE HOMOMORPHIC PROPERTIES AND COMPUTES:
C cm EncPK (≡¡ + [(K2)-1 ¡M˘ + (K2)-1 ¡R ¡X2 ¡X1 MOD q])
260
P2 SENDS C TO P1
265
P1 COMPUTES R = K1¡R2, S˘= DecSK (C) AND S = (K1)-1 ¡S˘MOD q
270
P1 VERIFIES THAT (R, S) IS A VALID SIGNATURE, OTHERWISE P1 ABORTS

Fig. 2

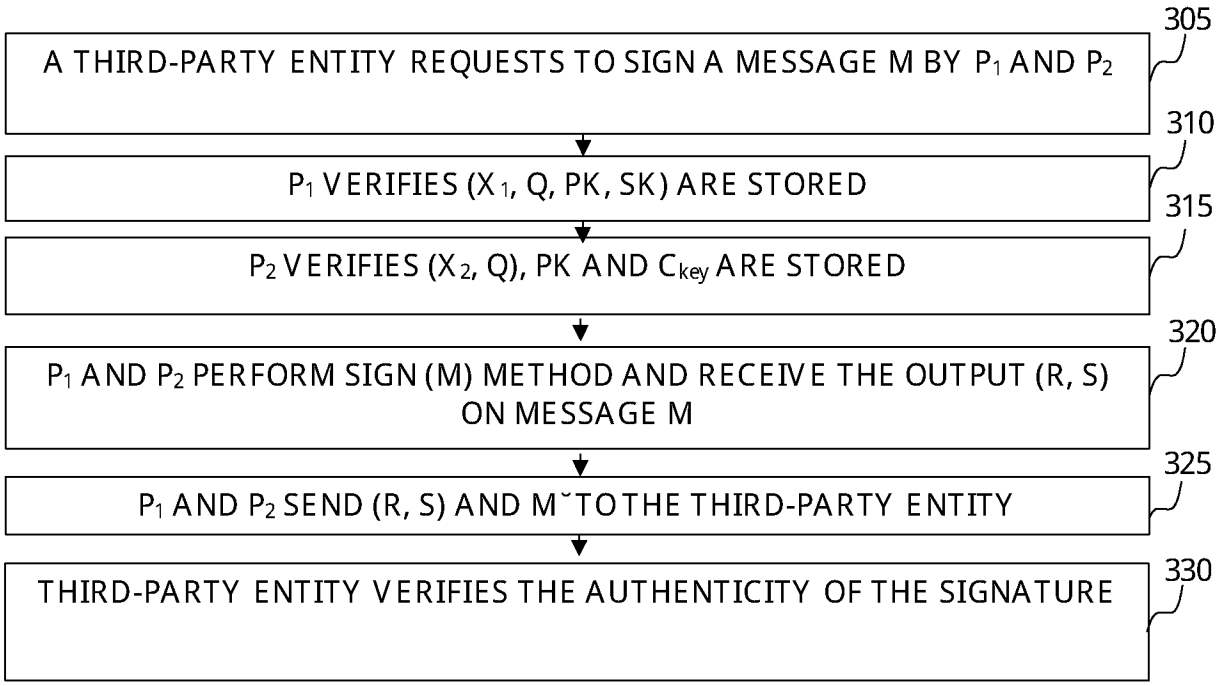
A THIRD-PARTY ENTITY REQUESTS TO SIGN A MESSAGE M BY P1 AND P2
305
P1 VERIFIES (X1, Q, PK, SK) ARE STORED
310
P2 VERIFIES (X2, Q), PK AND Ckey ARE STORED
315
P1 AND P2 PERFORM SIGN (M) METHOD AND RECEIVE THE OUTPUT (R, S) ON MESSAGE M
320
P1 AND P2 SEND (R, S) AND M˜TO THE THIRD-PARTY ENTITY
325
THIRD-PARTY ENTITY VERIFIES THE AUTHENTICITY OF THE SIGNATURE
330

Fig. 3

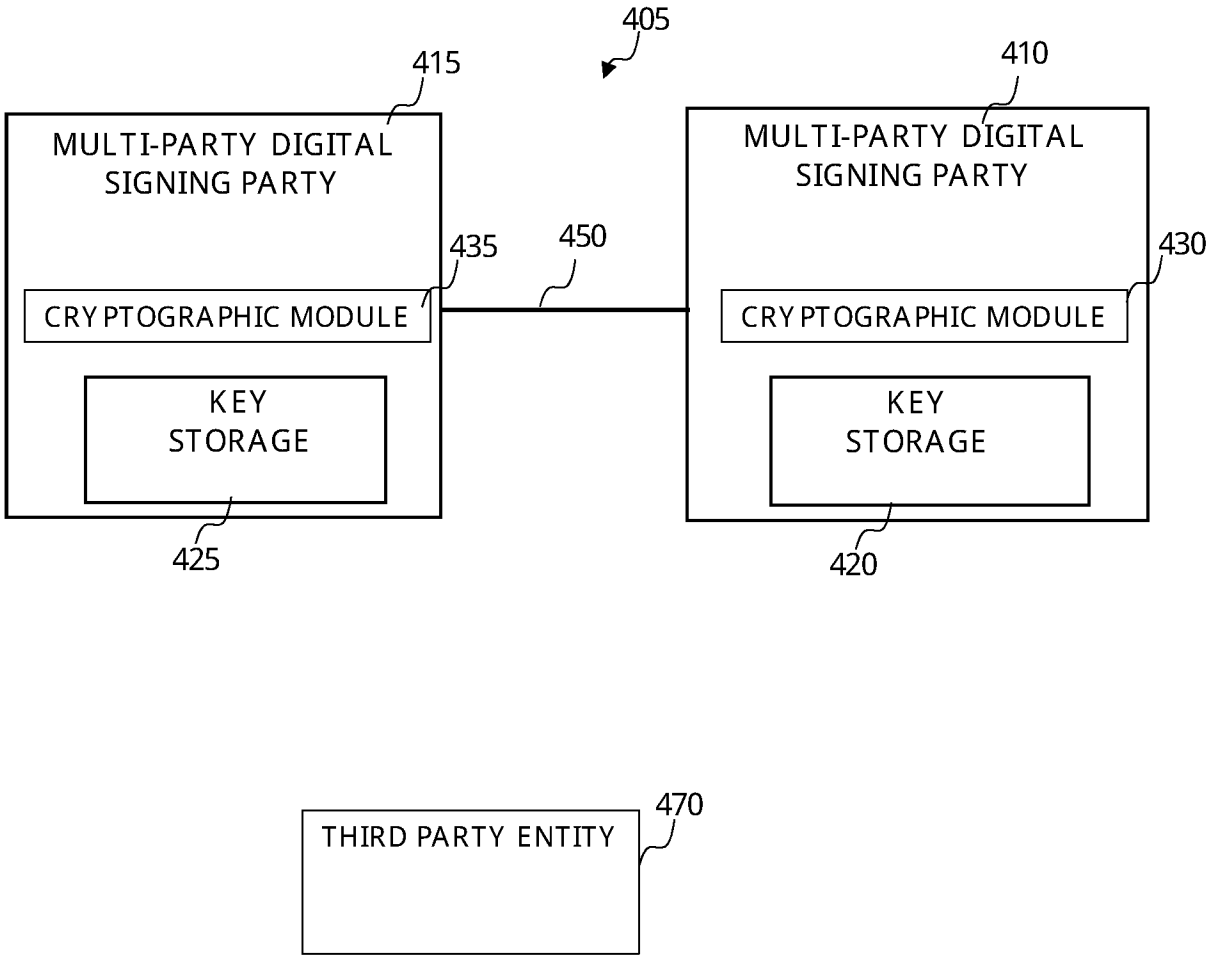
405
415
410
MULTI-PARTY DIGITAL SIGNING PARTY
MULTI-PARTY DIGITAL SIGNING PARTY
435
450
430
CRYPTOGRAPHIC MODULE
CRYPTOGRAPHIC MODULE
KEY STORAGE
KEY STORAGE
425
420
THIRD PARTY ENTITY
470

Fig. 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL2018/050505

A. CLASSIFICATION OF SUBJECT MATTER
IPC (2018.01) H04L 9/30, H04L 9/32, H04L 9/08

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)
IPC (2018.01) H04L 9/30, H04L 9/32, H04L 9/08

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)
Databases consulted: PATENTSCOPE, Esp@cenet, Google Patents, Orbit
Search terms used: digital sign, multiparty, ECDSA, Elliptical, SM2, homomorphic, lattice, paillier, BLISS, secret, shared, segmentation

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 9489522 B1 EL DEFRAWY et al. 08 Nov 2016 (2016/11/08) abstract, column 7 lines 10-22, column 8 lines 60-64, column 9 line 50 - column 10 line 2	1-12
Y	CN 106549770 A UNIV. WUHAN TECH 29 Mar 2017 (2017/03/29) Page 6, 7 lines 1-16, page 7 line 17 - page 8 line 7	1-12
Y	US 2002057796 A1 LAMBERT et al. 16 May 2002 (2002/05/16) ¶ 0024, 0068, 0102	8,11
Y	US 2016344557 A1 CHABANNE et al. 24 Nov 2016 (2016/11/24) ¶ 0060, 0063	9,12
P,X	CN 107707358 A UNIV WUHAN 16 Feb 2018 (2018/02/16) Whole document	1,10

[X] Further documents are listed in the continuation of Box C.

[X] 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
19 Aug 2018	20 Aug 2018
Name and mailing address of the ISA: Israel Patent Office Technology Park, Bldg.5, Malcha, Jerusalem, 9695101, Israel Facsimile No. 972-2-5651616	Authorized officer COPPENHAGEN Uri Telephone No. 972-2-5657811

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL2018/050505

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 7936869 B2 BEESON 03 May 2011 (2011/05/03) Whole document	1-7,10
A	"An Efficient Blind Digital Signature Protocol Based on Elliptic Curve." INTERNATIONAL JOURNAL OF TECHNOLOGY ENHANCEMENTS AND EMERGING ENGINEERING RESEARCH, VOL 2, ISSUE 9 62 ISSN 2347-42 (retrieved on 18-08-2018). Retrieved from the Internet:<https://pdfs.semanticscholar.org/7c9d/5b6b9add599c24c95a7861a66325429e1f4b.pdf> BARAI, S. 31 Dec 2014 (2014/12/31) Chapter 1	1-12
A	"SM2 Digital Signature Algorithm draft-shen-sm2-ecdsa-01. (retrieved on 19-08-2018) Retrieved from the Internet: < https://tools.ietf.org/html/draft-shen-sm2-ecdsa-02> SHEN, S. et al. 14 Feb 2014 (2014/02/14) Whole document	1-12
A	"Generating keys in elliptic curve cryptosystems." ISSN: 1694-2108 | Vol. 4, No. 1.(retrieved on 19-08-2018). Retrieved from the Internet:<https://pdfs.semanticscholar.org/3cd2/5456b5e12b21b8263996fbd712a2971adc47.pdf> VIDAKOVIC D. et al. 01 Aug 2013 (2013/08/01) Whole document	1-12
A	"Beyond ECDSA and RSA: Lattice-based Digital Signatures on Constrained Devices." (retrieved on 19-08-2018). Retrieved from the Internet:< https://www.sha.rub.de/media/attachments/files/2014/06/bliss_arm.pdf> ODER, T. et al. 05 Jun 2014 (2014/06/05) Whole document	8,11

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/IL2018/050505

Patent document cited search report	Publication date	Patent family member(s)	Publication Date
US 9489522 B1	08 Nov 2016	US 9489522 B1	08 Nov 2016
		CN 106664205 A	10 May 2017
		CN 108028751 A	11 May 2018
		EP 3132560 A1	22 Feb 2017
		EP 3132560 A4	20 Dec 2017
		US 9443089 B1	13 Sep 2016
		US 9449177 B1	20 Sep 2016
		US 9450938 B1	20 Sep 2016
		US 9467451 B1	11 Oct 2016
		US 9536114 B1	03 Jan 2017
		US 9558359 B1	31 Jan 2017
		US 9614676 B1	04 Apr 2017
		US 9787472 B1	10 Oct 2017
		US 2017317820 A1	02 Nov 2017
		WO 2015160839 A1	22 Oct 2015
		WO 2017075609 A1	04 May 2017
CN 106549770 A	29 Mar 2017	CN 106549770 A	29 Mar 2017
US 2002057796 A1	16 May 2002	US 2002057796 A1	16 May 2002
		AU 1851500 A	31 Jul 2000
		AU 1851700 A	31 Jul 2000
		CA 2257008 A1	24 Jun 2000
		CA 2257008 C	11 Dec 2007
		CA 2353395 A1	06 Jul 2000
		CA 2353395 C	08 Jul 2014
		CA 2354431 A1	06 Jul 2000
		DE 69903854 D1	12 Dec 2002
		DE 69903854 T2	31 Jul 2003
		EP 1140999 A1	10 Oct 2001
		EP 1141820 A1	10 Oct 2001

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/IL2018/050505

Patent document cited search report	Publication date	Patent family member(s)	Publication Date
		EP 1141820 B1	06 Nov 2002
		JP 2002533787 A	08 Oct 2002
		JP 4662577 B2	30 Mar 2011
		JP 2003512017 A	02 Apr 2003
		US 2002044649 A1	18 Apr 2002
		US 7110538 B2	19 Sep 2006
		US 7297341 B1	20 Nov 2007
		US 2008095793 A1	24 Apr 2008
		US 7553493 B2	30 Jun 2009
		US 2009280168 A1	12 Nov 2009
		US 7850980 B2	14 Dec 2010
		US 2006029222 A1	09 Feb 2006
		US 7995752 B2	09 Aug 2011
		US 2002037293 A1	28 Mar 2002
		US 2008112965 A1	15 May 2008
		WO 0039158 A1	06 Jul 2000
		WO 0039668 A1	06 Jul 2000
US 2016344557 A1	24 Nov 2016	US 2016344557 A1	24 Nov 2016
		EP 3091689 A1	09 Nov 2016
		EP 3091689 B1	21 Mar 2018
		FR 3035986 A1	11 Nov 2016
		FR 3035986 B1	27 Jul 2018
CN 107707358 A	16 Feb 2018	CN 107707358 A	16 Feb 2018
US 7936869 B2	03 May 2011	US 2006153366 A1	13 Jul 2006
		US 7936869 B2	03 May 2011
		US 2006156012 A1	13 Jul 2006
		US 7490239 B2	10 Feb 2009
		US 2006153365 A1	13 Jul 2006

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/IL2018/050505

Patent document cited search report	Publication date	Patent family member(s)	Publication Date
		US 7593527 B2	22 Sep 2009
		US 2006153371 A1	13 Jul 2006
		US 7693277 B2	06 Apr 2010
		US 2006153368 A1	13 Jul 2006
		US 7869593 B2	11 Jan 2011
		US 2006153364 A1	13 Jul 2006
		US 2006153367 A1	13 Jul 2006
		US 2006153369 A1	13 Jul 2006
		US 2006153370 A1	13 Jul 2006
		US 2006156013 A1	13 Jul 2006