

(51) International Patent Classification:
H04L 9/30 (2006.01)(21) International Application Number:
PCT/IB2012/052930(22) International Filing Date:
10 June 2012 (10.06.2012)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
213497 12 June 2011 (12.06.2011) IL
216806 7 December 2011 (07.12.2011) IL(71) Applicant (for all designated States except US): **NDS LIMITED** [GB/GB]; One London Road, Staines Middlesex TW18 4EX (GB).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **KIPNIS, Aviad** [IL/IL]; P.o.b. 6678, 90435 Efrat (IL). **HIBSHOOSH, Eliphaz** [IL/IL]; 5 Sirkin Street, 63562 Tel Aviv (IL).(74) Agents: **KATZ, Samuel M.** et al.; NDS Legal (Patents), One London Road, Staines Middlesex TW18 4EX (GB).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, 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, 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, ML, MR, NE, SN, TD, TG).

Published:

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

[Continued on next page]

(54) Title: PUBLIC KEY CRYPTOGRAPHY WITH REDUCED COMPUTATIONAL LOAD

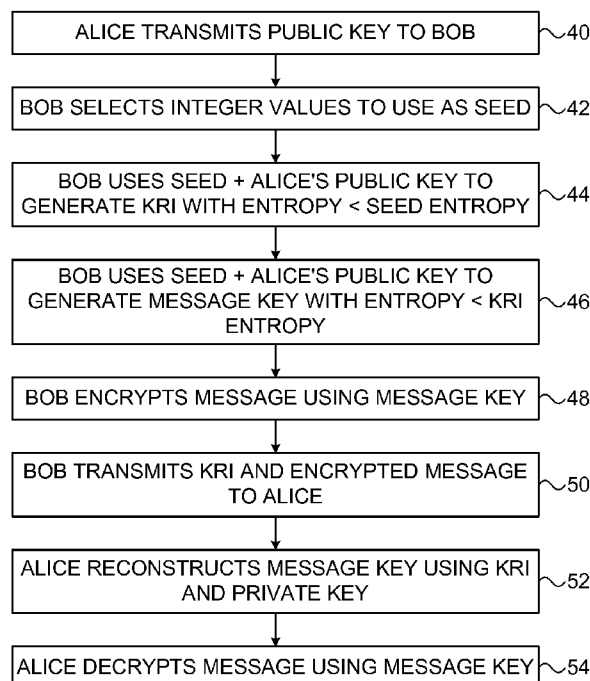


FIG. 2

(57) Abstract: A cryptographic method, comprising receiving a public key belonging to a message recipient having a private key corresponding to the public key, selecting a numerical seed, having a first entropy, using the numerical seed and the public key, generating key recovery information having a second entropy, which is less than the first entropy, and generating a message key having a third entropy, which is less than the first entropy, encrypting a message using the message key, and transmitting the encrypted message and the key recovery information to the message recipient, whereby the recipient reconstructs the message key using the key recovery information and the private key and decrypts the encrypted message using the reconstructed message key. Related systems and methods are also included.



-
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

PUBLIC KEY CRYPTOGRAPHY WITH REDUCED COMPUTATIONAL LOAD**FIELD OF THE INVENTION**

The present invention relates generally to cryptography, and particularly to systems and methods for
5 secure exchange of cryptographic keys.

BACKGROUND

Public-key cryptographic techniques are widely used for encryption and authentication of electronic documents. Such techniques use a mathematically-related
10 key pair: a secret private key and a freely-distributed public key. A sender encrypts a message using the recipient's public key. The recipient then decrypts the message using the corresponding private key. In a well-designed scheme, it is computationally infeasible to
15 reconstruct the private key from the public key or to otherwise decrypt the message without having possession of the private key.

Commonly-used public-key cryptographic techniques, such as the Rivest Shamir Adleman (RSA) algorithm, rely
20 on numerical computations over large finite fields. To ensure security against cryptanalysis, these techniques require the use of large keys and complex computations, which are costly, in terms of memory and computing power, to store and compute. These demands can be problematic
25 in application environments such as smart cards, in which computing resources are limited.

Various alternative public-key signature schemes have been developed in order to reduce the resource burden associated with cryptographic operations. Some
30 schemes of this sort are based on the Merkle-Hellman knapsack cryptosystem, which was first described by Merkle and Hellman in "Hiding Information and Signatures

in Trapdoor Knapsacks," *IEEE Transactions on Information Theory* **24:5**, pages 525-530 (1978). Unfortunately, knapsack-based schemes, including those based on more secure iterated-knapsack approaches that were developed
5 subsequently, have been found to be vulnerable to attacks based on lattice reduction and the Diophantine approximation. Such vulnerabilities are reviewed, for example, by Brickell in "Breaking Iterated Knapsacks," *Advances in Cryptology - Crypto '84*, pages 342-358
10 (Springer-Verlag, 1985).

SUMMARY

Embodiments of the present invention that are described hereinbelow provide methods and systems for public key cryptography that can be implemented at
15 relatively low computational cost but are secure against known attacks.

There is therefore provided, in accordance with an embodiment of the present invention, a cryptographic method, which includes receiving a public key belonging
20 to a message recipient having a private key corresponding to the public key. A numerical seed, having a first entropy, is selected. Using the numerical seed and the public key, key recovery information is generated, having a second entropy, which is less than the first entropy,
25 and a message key is generated, having a third entropy, which is less than the first entropy. A message is encrypted using the message key. The encrypted message and the key recovery information are transmitted to the message recipient, whereby the recipient reconstructs the
30 message key using the key recovery information and the private key and decrypts the encrypted message using the reconstructed message key.

In the disclosed embodiments, generating the key recovery information and generating the message key include performing linear algebraic operations. Typically, the key recovery information and the message key are generated and the message key is reconstructed without performing exponentiation.

In some embodiments, the numerical seed includes a set of integer values, the public key includes a matrix of first integer entries and a vector of second integer entries, and generating the message key includes multiplying the integer values by the vector, and generating the key recovery information includes multiplying the integer values by the matrix. The method may include generating the public key by iteratively performing modulo operations and multiplications using elements of the private key.

Additionally or alternatively, the method includes reconstructing the message key using linear algebraic operations. Reconstructing the message key may include iteratively performing modulo operations and multiplications using the key recovery information and elements of the private key.

Alternatively, the public key is derived from a first set of hidden linear variables, including vectors in a finite field, and a set of error vectors in the finite field, and generating the key recovery information includes computing a key recovery vector and an error correcting code using the public key. The method may include reconstructing the message key using the private key, while performing an error correction computation using the error correcting code.

Typically, encrypting the message includes applying a symmetric cipher to the message using the message key.

In a disclosed embodiment, the third entropy is less than the second entropy.

5 In one embodiment, generating the key recovery information and the message key includes generating first key recovery information and a first message key. Encrypting the message includes receiving second key recovery information from the message recipient,
10 reconstructing a second message key using the second key recovery information, and encrypting the message using the first and second message keys together.

 There is also provided, in accordance with an embodiment of the present invention, a cryptographic
15 method, which includes holding a private key at a message recipient, while a public key corresponding to the private key is provided to a sender. Key recovery information, computed by the sender using the public key and a selected numerical seed, is received from the
20 sender. The numerical seed is not transmitted to the message recipient and has a first entropy, while the key recovery information has a second entropy, which is less than the first entropy. A message, which is encrypted using a message key computed by the sender using the
25 integer values and the public key, is received from the sender, wherein the message key is not transmitted to the message recipient and has a third entropy, which is less than the first entropy. The message key is reconstructed at the message recipient using the key recovery
30 information and the private key. The encrypted message is decrypted using the reconstructed message key.

Typically, reconstructing the message key includes applying linear algebraic operations to the key recovery information using elements of the private key, without performing exponentiation.

5 In one embodiment, reconstructing the message key includes iteratively performing modulo operations and multiplications using the key recovery information and the elements of the private key.

10 In another embodiment, the key recovery information includes an error correcting code, and reconstructing the message key includes performing an error correction computation using the error correcting code. A key recovery information vector may be multiplied by a matrix to generate an approximate message key, to which the
15 error correction computation is applied in order to reconstruct the message key.

There is additionally provided, in accordance with an embodiment of the present invention, cryptographic apparatus, including a memory, which is configured to
20 hold a public key belonging to a message recipient having a private key corresponding to the public key. A processor is configured to select a numerical seed having a first entropy, to use the numerical seed and the public key in generating key recovery information having a
25 second entropy, which is less than the first entropy, and in generating a message key having a third entropy, which is less than the first entropy, to encrypt a message using the message key, and to transmit the encrypted message and the key recovery information to the message
30 recipient, whereby the recipient reconstructs the message key using the key recovery information and the private

key and decrypts the encrypted message using the reconstructed message key.

There is further provided, in accordance with an embodiment of the present invention, cryptographic apparatus, including a memory, which is configured to hold a private key and a public key corresponding to the private key. A processor is configured to transmit the public key to a sender, and to receive from the sender key recovery information, which is computed by the sender using the public key and a numerical seed, and a message, which is encrypted using a message key computed by the sender using the integer values and the public key. The numerical seed and the message key are not transmitted by the sender, and the numerical seed has a first entropy, while the key recovery information has a second entropy, which is less than the first entropy, and the message key has a third entropy, which is less than the first entropy. The processor is configured to reconstruct the message key using the key recovery information and the private key, and to decrypt the encrypted message using the reconstructed message key.

There is moreover provided, in accordance with an embodiment of the present invention, a computer software product, including a computer-readable medium in which program instructions are stored, which instructions, when read by a processor, cause the processor to receive a public key belonging to a message recipient having a private key corresponding to the public key, to select a numerical seed having a first entropy, to use the numerical seed and the public key in generating key recovery information having a second entropy, which is less than the first entropy, and in generating a message

key having a third entropy, which is less than the first entropy, to encrypt a message using the message key, and to transmit the encrypted message and the key recovery information to the message recipient, whereby the
5 recipient reconstructs the message key using the key recovery information and the private key and decrypts the encrypted message using the reconstructed message key.

There is furthermore provided, in accordance with an embodiment of the present invention, a computer software
10 product, including a computer-readable medium in which program instructions are stored, which instructions, when read by a processor, cause the processor to access a memory holding a private key and a public key corresponding to the private key, to receive from a
15 sender key recovery information, which is computed by the sender using the public key and a numerical seed, and a message, which is encrypted using a message key computed by the sender using the integer values and the public key. The numerical seed and the message key are not
20 transmitted by the sender, and the numerical seed has a first entropy, while the key recovery information has a second entropy, which is less than the first entropy, and the message key has a third entropy, which is less than the first entropy. The instructions cause the processor
25 to reconstruct the message key using the key recovery information and the private key, and to decrypt the encrypted message using the reconstructed message key.

The present invention will be more fully understood from the following detailed description of the
30 embodiments thereof, taken together with the drawings in which:

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a schematic, pictorial illustration of a cryptographic system, in accordance with an embodiment of the present invention;

5 Fig. 2 is a flow chart that schematically illustrates a method for secure communication, in accordance with an embodiment of the present invention;

Fig. 3 is a block diagram that schematically illustrates a method for encryption of a message, in
10 accordance with an embodiment of the present invention;

Fig. 4 is a block diagram that schematically illustrates a method for decryption of a message, in accordance with an embodiment of the present invention; and

15 Fig. 5 is a flow chart that schematically illustrates a method for secure communication, in accordance with another embodiment of the present invention.

20 DETAILED DESCRIPTION OF EMBODIMENTS

OVERVIEW

Public key cryptographic methods that are in common use, such as RSA, typically require the encrypting and decrypting processors to perform multiple exponentiation
25 operations over large bases and exponents. This sort of mathematical approach offers high security but requires substantial computational power to implement (since even a single exponentiation of this sort is difficult). Such methods may therefore be impractical for implementation

in low-cost and power-limited devices, such as smart cards and radio frequency identification (RFID) tags.

Embodiments of the present invention that are described hereinbelow provide methods for public key cryptography that can be implemented using only linear algebraic operations (i.e., multiplication and addition), without performing exponentiation. These methods are therefore suitable particularly for use in low-cost and power-limited devices. Some of these embodiments apply computational methods similar to those used in iterated knapsack cryptography, but with novel enhancements that provide resistance against known attacks while maintaining a low computational load. These embodiments can thus make public key cryptography practical even for low-performance devices.

In the disclosed embodiments, a message originator (also referred to as the "sender") receives a public key belonging to the intended message recipient. The recipient maintains a secret private key, from which the public key is derived. To encrypt the message, the sender selects a secret numerical seed (which can be a string of bits or a set of integer values, for example), which may be chosen at random. The sender uses the seed value together with the recipient's public key to generate a message key and also to generate key recovery information. The sender encrypts the message using the message key, by means of a symmetric encryption algorithm (also referred to as a symmetric cipher). Any suitable symmetric cryptographic algorithm, such as Advanced Encryption Standard (AES) encryption, may be used for this purpose, and the choice of this algorithm is beyond the scope of the present invention.

The sender transmits the encrypted message and the key recovery information to the recipient. The recipient reconstructs the message key using the key recovery information and the recipient's own private key. The recipient is then able to decrypt the encrypted message using the reconstructed message key with the same symmetric cryptographic algorithm that was used for encryption.

Methods that can be used to derive the public key from the private key, and then to generate the message key and key recovery information using the public key, are described in detail hereinbelow. The security of the cryptographic scheme that is based on these methods stems, at least in part, from the manner in which the seed, the key recovery information, and the message key itself are related to one another in terms of entropy. In information theory, entropy is a measure of the uncertainty associated with a random variable. In the context of the present patent application and in the claims, the term "entropy" refers to Shannon entropy, which quantifies the expected value of the information contained in a message (which can itself be considered a specific realization of a random variable), usually in units such as bits. When the message is a random binary string, the entropy is given by the number of bits in the string.

In the disclosed embodiments, the seed has greater entropy than the key recovery information, which in turn has greater entropy than the message key. As a result, an attacker cannot practically use the key recovery information to recover the seed; and the message key itself cannot practically be used to recover either the

key recovery information or the seed. Therefore, the scheme is secure despite its computational simplicity.

SYSTEM DESCRIPTION

Fig. 1 is a schematic, pictorial illustration of a cryptographic system 20, in accordance with an embodiment of the present invention. In the pictured embodiment, a server 22 exchanges secret information with a client device 24. The server in this example may be a general-purpose computer, comprising a processor 25 and a memory 26, with an interface 28 for communicating with device 24. Interface 28 is pictured as comprising a wireless antenna, but other types of wireless and/or wired links may be provided alternatively or additionally for communicating with clients. The server may also comprise a display 30 and/or other peripheral components. Typically, server 22 performs the functions that are described herein under the control of software. This software may be downloaded to the server in electronic form, over a network, for example. Alternatively or additionally, the software may be held on tangible, non-transitory computer-readable media, such as optical, magnetic, or electronic memory components.

Client device 24 in this example comprises a smart card, comprising a processor 32, a memory 34, and a communication interface 36, and possibly other components such as a display 38. As in the case of interface 28, interface 36 is shown in the figure as comprising a wireless antenna, but may alternatively comprise any other suitable sort of wireless or wired link. Processor 32 is typically programmed in software or firmware, which may be stored in memory 34, to carry out the functions

that are described herein. Alternatively or additionally, processor 32 may comprise dedicated hardware logic for carrying out at least some of these functions. (In view of the computational simplicity of the cryptographic methods that are described hereinbelow, however, it is generally unnecessary that device 24 comprise dedicated cryptographic accelerator circuits, as in some devices that are known in the art.)

Upon establishing a connection with server 22, device 24 transmits its public key to the server. The corresponding private key is held securely in memory 34 and is not transmitted or otherwise disclosed outside device 24. Based on the public key and a seed, which may be chosen at random, processor 25 computes a dedicated message key and key recovery information (KRI), as described hereinbelow. The message key is held securely in memory 26 and is not transmitted or otherwise disclosed explicitly outside server 25. Server 22 applies the message key to encrypt a message for transmission to device 24, using a suitable symmetric encryption algorithm, and then transmits the encrypted message (as indicated by ENC(MSG) in the figure) and the KRI to device 24. Processor 32 uses the KRI together with the private key held in memory 34 to recover the message key, and then applies the message key in decrypting the message. Further messages may be transmitted subsequently using the same key or using a different key, which may be generated using the same public key with a new seed.

Device 24 may likewise receive a public key belonging to server 22 and may use it to generate a message key and KRI in a similar fashion. The algorithms

both for generation of the message key and KRI and for key recovery are all computationally sufficiently lightweight to be carried out by processor 32. As another alternative, both server 22 and device 24 may
5 generate respective message keys and transmit corresponding key recovery information to one another, following which messages are encrypted and decrypted using both message keys together. This sort of alternative is described in greater detail hereinbelow
10 with reference to Fig. 5.

For the sake of clarity and convenience, embodiments of the present invention are described hereinbelow by way of example with reference to the elements of system 20. The principles of the present invention are by no means
15 limited to this sort of system, however, and may alternatively be implemented in substantially any sort of cryptographic environment, including both client-server and peer-to-peer applications, and including substantially any and all sorts of computing and
20 cryptographic devices.

Fig. 2 is a flow chart that schematically illustrates a method for secure communication, in accordance with an embodiment of the present invention. The participants in the method of Fig. 2 are a pair of
25 computing devices, which are identified, for the sake of convenience, as "Bob" and "Alice." For example, "Bob" may identify server 22, while "Alice" identifies device 24.

As an initial step (as a stage in the manufacture of
30 device 24, for example), a private key is chosen for Alice and is stored securely in memory 34; typically, the private key is never transmitted or disclosed to any

entity outside device 24. In an iterated scheme that is described in detail hereinbelow, the private key may comprise several integer vectors, which are selected at random subject to certain constraints, as described hereinbelow. The public key, which may comprise a $m \times n$ matrix and a vector of n integers, is computed from the private key in an iterative process, also described hereinbelow, and is likewise stored in memory 34. In an alternative scheme based on hidden linear variables, which is also described below, the private and public keys comprise other sorts of matrices, but the general principles of operation, as illustrated in Fig. 2, are substantially the same. Upon establishing communications with Bob, Alice transmits her public key to Bob, at a public key transmission step 40. Alternatively, Bob could receive Alice's public key from an authorized database or other authority, rather than from Alice directly. The public key is typically stored in memory 26, along with other values used by processor 24 in the course of computations.

In accordance with the iterative scheme, Bob selects a vector of n integer values to use as a seed in generating a message key and KRI, at a seed selection step 42. These integer values may be selected at random and typically have a length, in bits, that is equal to (or possibly greater than) the intended length L of the message key. (If these values have more than L bits, then the key recovery process will not result in the proper message key.)

Typically, to protect against brute-force attempts to discover the message key, $L \geq 64$. Optionally, for greater security, L may be at least 100, or possibly 128

or more. Bob applies the seed vector, together with the matrix portion of Alice's public key, in generating the KRI, at a KRI generation step 44. For example, the $m \times n$ matrix portion of the public key may be multiplied by the
5 n -element seed vector to give a vector of m KRI elements. To ensure security of the scheme, the values of m and n are typically chosen with $m < n$, so that the KRI vector has fewer elements, and thus lower entropy, than the seed vector. The inventors have found it advantageous to
10 choose $m < n/2$, so that the seed vector has substantially more entropy than the KRI, in order to foil attacks that are based on lattice reduction techniques, as are known in the art. In the three-iteration scheme, m may be either 1 or 2, so that the KRI contains only one or two
15 elements, although larger values of m may also be used.

Bob uses the same seed vector, together with the vector portion of Alice's public key, to generate the message key, at a message key generation step 46. The message key may, for example, be an integer value of
20 length L , which is computed by vector multiplication between the vector portion of the public key and the seed vector. The message key thus has lower entropy than both the seed and the KRI. Bob encrypts a message to Alice using the message key, at a message encryption step 48.
25 As noted earlier, Bob may use any suitable symmetric cryptographic algorithm at this step, such as AES. Bob then transmits the encrypted message and the KRI computed at step 44 to Alice, at a transmission step 50.

Alice receives the message and KRI from Bob, and
30 then uses the KRI together with elements of her private key to recover the message key, at a key recovery step 52. The key recovery computation involves iterative

modular multiplications, as described in detail hereinbelow. The number of iterations (which is equal to the number of iterations used in generating the public key from the private key) is determined in advance, depending on the desired security level: Using more iterations gives greater security, but also requires larger keys and KRI and increases the transmission bandwidth required and the computational burden involved in encryption and decryption. The inventors have found that two or three iterations are sufficient for most practical applications. Typical parameters for two- and three-iteration schemes are listed below.

Alice uses the message key that was reconstructed at step 52 in decrypting the encrypted message from Bob, at a decryption step 54. At this step, Alice applies the same symmetric cryptographic algorithm that was used at step 48.

ALGORITHMIC DETAILS - ITERATED CRYPTOGRAPHIC SCHEME

Fig. 3 is a block diagram that schematically illustrates a method for encryption of a message, in accordance with an embodiment of the present invention. This figure and the accompanying description below provide details of a possible implementation of steps 40-48 in Fig. 2.

Alice's private key in a two-iteration version of this scheme has the following components:

1. $P_1, \dots, P_m$ - m large prime numbers.
2. Elements $\alpha_1, \dots, \alpha_m$, wherein each α_i is an element modulo P_i .

3. $Q_1, \dots, Q_m$ - m large prime numbers, such that $Q_i < P_i$, but Q_i is close to P_i (for example, the ten most significant bits of Q_i and P_i are identical).

4. Elements $\beta_1, \dots, \beta_m$, wherein each β_i is an element modulo

5 Q_i satisfying: $2^L \mid (-\beta_i^{-1} \cdot P_i) \bmod Q_i < \frac{1}{2n} \cdot \left\lfloor \frac{Q_i}{2^L} \right\rfloor$

(meaning that the term to the right of the " $\mid$ " is divisible by the term to the left, in this case 2^L).

Here and below, the inverse (-1 power) operation refers to the modular inverse ($\bmod Q_i$ in this case), and n is

10 a parameter of the scheme as defined above.

In the three-iteration version the scheme, the components of the private key are as follows:

1. $P_1, \dots, P_m$, $\alpha_1, \dots, \alpha_m$, and $Q_1, \dots, Q_m$ are defined as in the two iteration scheme above.

15 2. Elements $\beta_1, \dots, \beta_m$, wherein each β_i is an element modulo

Q_i satisfying: $(-\beta_i^{-1} \cdot P_i) \bmod Q_i < \frac{1}{3n} \cdot \left\lfloor \frac{T_i}{2^L} \right\rfloor$.

3. $T_1, \dots, T_m$ - m large prime numbers, such that $T_i < Q_i$ but T_i is close to Q_i (for example, the ten most significant bits of Q_i and T_i are identical).

20 4. Elements $\gamma_1, \dots, \gamma_m$, wherein each γ_i is an element modulo T_i satisfying:

$2^L \mid (\gamma_i^{-1} \cdot (-\beta_i^{-1} \cdot P_i \bmod Q_i) \bmod T_i) < \frac{1}{3n} \cdot \left\lfloor \frac{T_i}{2^L} \right\rfloor$ and

$2^L \mid (-\gamma_i^{-1} \cdot Q_i) \bmod T_i < \frac{1}{3n} \cdot \left\lfloor \frac{T_i}{2^L} \right\rfloor$. (Here the inverse (-1 power) operation on γ_i refers to the modular inverse

25 $\bmod T_i$.)

Alice's public key comprises a matrix component 62 and a vector component 64, defined as follows:

- Matrix A comprises $m \times n$ integer values $a_{i,j}$ in the range $[0 \dots N]$ (wherein N is a parameter of the scheme), arranged in m rows and n columns.
- Vector B comprises n integer entries b_j in the range $[0 \dots 2^L]$.

To generate the public key components from the private key, a secret matrix X of dimensions $m \times n$ is selected. In the two-iteration scheme, the elements of the i 'th row of the matrix X are numbers modulo Q_i , and thus the values $X_{i,j}$ may be regarded as integers in the range $0 \leq X_{i,j} < Q_i$. The elements $X_{i,j}$ are required to satisfy the following constraints: $X_{i,j} < \frac{1}{2n} \cdot \left\lfloor \frac{Q_i}{2^L} \right\rfloor$.

The public key elements $a_{i,j}$ of matrix A (component 62) are computed by iterative modular vector multiplication: $a_{i,j} = (\alpha_i \cdot (X_{i,j} \cdot \beta_i \bmod Q_i)) \bmod P_i$. The public key elements b_j of vector B (component 64) are computed by summation over the elements of the matrix X :

$$b_j = \left(\sum_{i=1}^m X_{i,j} \right) \bmod 2^L.$$

In the three-iteration scheme the computation of the public key components is similar, but the values $X_{i,j}$ are numbers modulo T_i and can thus be regarded as integers in the range $0 \leq X_{i,j} < T_i$. The elements $X_{i,j}$ are required to satisfy $X_{i,j} < \frac{1}{3n} \cdot \left\lfloor \frac{T_i}{2^L} \right\rfloor$. The elements of matrix A are defined as:

$a_{i,j} = (\alpha_i \cdot (\beta_i \cdot (X_{i,j} \cdot \gamma_i \bmod T_i) \bmod Q_i) \bmod P_i)$, while the elements of vector B are the same as in the two-iteration scheme.

Seed 60 comprises a vector of n random integer values $K_1, \dots, K_n$ in the range $0 \leq K_i < 2^L$. In key generation 66, a message key (MK) 68 is generated by vector multiplication between vector component 64 of the public key and seed 60, followed by a modulo operation:

$$MK = (\sum_{j=1}^n b_j \cdot K_j) \bmod 2^L.$$
 Key 68 is used in a standard symmetric cipher (encryption algorithm) 72 to turn Bob's message 70 into an encrypted message 74 for transmission to Alice.

In KRI generation 76, matrix component 62 of the public key is multiplied by seed 60 to give KRI 78, comprising a vector of m integers, in which each element is given by $KRI_i = \sum_{j=1}^n a_{i,j} \cdot K_j$; $1 \leq i < m$.

Fig. 4 is a block diagram that schematically illustrates a method for decryption of message 74, in accordance with an embodiment of the present invention. This figure and the accompanying description below provide details of a possible implementation of steps 52-54 in Fig. 2. Alice applies KRI 78 and her stored private key 80 (from memory 34) to compute the value of message key 68, in key recovery 82.

The computation here, too, involves iterative modular vector multiplications: In the two-iteration scheme Alice calculates interim vector elements $Y_i = [(\beta_i^{-1} \cdot (KRI_i \cdot \alpha_i^{-1}) \bmod P_i) \bmod Q_i] \bmod 2^L$, and then performs summation and modulo operations over these

components to compute message key 68: $MK = (\sum_{i=1}^m Y_i) \bmod 2^L$

. For the three-iteration scheme Alice calculates the vector elements Y_i as follows:

$$Y_i = (\gamma_i^{-1} \cdot (\beta_i^{-1} \cdot (\alpha_i^{-1} \cdot KRI_i \bmod P_i) \bmod Q_i) \bmod T_i) \bmod 2^L,$$

5 and then sums the elements as in the two-iteration scheme to find key 68.

Alice applies key 68 to encrypted message 74 in a standard decipher 84, corresponding to cipher 72, and thus decrypts Bob's original message 70.

10 The inventors have evaluated the encryption and decryption scheme described above and found it to be correct and secure against attack. Although the present scheme uses principles that in some ways resemble "iterated knapsack" encryption schemes, it avoids the
15 vulnerabilities inherent in the type of trapdoor that is normally used in such schemes. Rather, by appropriate choice of parameters, to give sufficient differentiation of entropy between the secret seed and private key elements - which are never transmitted - and the KRI and
20 message key, the present scheme is highly resistant to known types of attack, including attacks based on finding simultaneous Diophantine approximations, despite the low computational complexity of the scheme.

The parameters of the encryption scheme described
25 above can be selected based on a tradeoff between security and computational burden, which is a function of the data size and the computational complexity. As noted earlier, a message key length of $L = 100$ bits is typically sufficient to foil brute force decryption
30 attacks, and it will therefore be assumed that $L = 100$. In the two-iteration scheme, it is desirable that the

size N of the integer elements of matrix A , which depends on the size of the elements of the private key, be (in binary terms) at least twice the length of the message key. In other words, N is at least 200 bits long, and could reasonably be set to 400 bits, so that $\log_2(N) = 400$. In this case, it can be shown that the seed vector should have $n = 6$ elements. With this choice of parameters, the public key size is about 3 Kbits, and the KRI size about 500 bits.

For greater security, the parameters could be chosen to be $\log_2(N) = 1000$, $L = 100$, $n = 16$. In this case, the public key size is about 17.6 Kbits, and the KRI is about 1.1 Kbits. The complexity of encryption and decryption in this latter case is about 4-5 times greater than using the smaller parameters from the preceding example.

In the three-iteration scheme, the parameters may be set to $\log_2(N) = 500$, $L = 100$, $n = 13$; or for greater security to $\log_2(N) = 1000$, $L = 100$, $n = 26$. The first of these sets of parameters gives a public key size of about 14.3 Kbits and KRI size of 1.2 Kbits; while the second gives a public key size of about 54.6 Kbits and KRI size of 2.2 Kbits. The complexity of encryption and decryption is about four times greater for the second parameter set than for the first and is in either case several times higher than in the two-iteration scheme.

ALTERNATIVE EMBODIMENT - CRYPTOGRAPHIC SCHEME BASED ON HIDDEN LINEAR VARIABLES

The principles of the present invention that are set forth above may similarly be applied in other novel cryptographic schemes that are based on linear algebraic computations, without exponentiation. For example, in

the embodiment described below, the scheme is based on a set of hidden linear variables, in the form of m vectors $y_1, y_2 \dots y_m$ in Z_P^n , which are used in the construction of the public and private keys. (Z_P is the field of integers modulo P , and Z_P^n is the vector space of n -element vectors over Z_P .) The hidden linear variables may be arranged in m columns forming a $n \times m$ HLV matrix over Z_P .

In the present HLV-based scheme, the key recovery information (KRI) that is transmitted to the message recipient for use in reconstructing the message key comprises two components: a KRI vector KRI_v , and an error correcting code KRI_{ECC} . The public key in this scheme comprises three matrices:

- B - $n \times r$ matrix of mod P numbers, which is used for generation of the message key (MK), of size r . B comprises several special linear combinations of $y_1, y_2 \dots y_m$, as described below. "Bob" generates the message key using the matrix B and a numerical seed X , comprising a sequence of n bits, which may be chosen at random.
- A - $n \times m$ matrix of mod P numbers, which is used for generation of the KRI vector KRI_v . A comprises m linear combinations of the vectors $y'_1, y'_2 \dots y'_m$, wherein $y'_i = y_i + e_i$, and $\{e_i\}$ is a set of small, low-weight error vectors in Z_P^n . Each e_i has h entries that are 1, while all the remaining entries are 0 (wherein h is small relative to n). The set of error vectors may be arranged in a $n \times m$ matrix E, in which each column i contains the corresponding vector e_i . We can thus

write $Y' = HLV + E$, wherein Y' is the matrix of the perturbed variables y'_i .

- $C - r \times l$ matrix, with elements in Z_p , used for generation of the error correcting code KRI_{ECC} , wherein
5 l is the number of ECC equations. A possible choice of parameters that simplifies the ECC calculations is to set $l = r - 1$, and to construct the matrix C so that only elements $C_{i,i}$ and $C_{i+1,i}$ are non-zero.

The private key comprises:

- 10 • $D - a m \times r$ secret matrix used to transform the KRI vector KRI_v into an approximation MK' of the message key, which contains errors of a particular kind.
- $S - A$ set of coefficients used together with KRI_{ECC} in order to correct MK' and thus recover MK .
- 15 In the HLV-based scheme, as in the iterated scheme described earlier, the seed X has greater entropy than MK and than the KRI.

To generate the public key, two secret matrices, M_A and M_{AB} , are used, together with the :

- 20 • $M_A - a m \times m$ mixing matrix, which is regular with random entries in Z_p . M_A is used to generate the matrix A by mixing of the perturbed variables: $A = Y' \cdot M_A$.
- $M_B - a$ special $m \times r$ matrix, which is used to generate
25 matrix B by mixing of the hidden linear variables: $B = HLV \cdot M_B$. Each column of the matrix M_B contains $k = k_1 + k_2 + k_3$ non-zero elements, which are divided into three groups, each containing k_i elements, wherein all the elements in each group share the same
30 coefficient c_i . In each column of M_B the three

coefficients c_1, c_2, c_3 are selected independently and satisfy the criterion:

$$\min_{r \neq 0} \{ \max\{|r \cdot c_1|, |n - r \cdot c_1|, |r \cdot c_2|, |n - r \cdot c_2|, |r \cdot c_3|, |n - r \cdot c_3|, \} \} > P^{2/3}$$

The private key is also generated using these two
5 secret matrices:

- D is defined as $= M_A^{-1} \cdot M_B$.
- S is a list of l triplets, wherein triplet i contains the three coefficients in the i th column of M_B , which will be denoted $(S_{i,1}, S_{i,2}, S_{i,3})$.

10 After receiving Alice's public key (matrices A, B and C), Bob generates a message key (MK) for use in symmetric encryption of a message to Alice by multiplying the seed X by the matrix B: $MK = X \cdot B$. MK is thus a vector of r modulo P elements. The KRI vector has m
15 modulo P elements, computed by Bob as $KRI_v = X \cdot A$. Bob computes KRI_{ECC} as a vector of l modulo P elements, given by $KRI_{ECC} = MK \cdot C$. Bob encrypts his message using MK, and then transmits the encrypted message together with the key recovery information KRI_v and KRI_{ECC} to Alice.

20 Upon receiving the encrypted message and KRI from Bob, Alice computes $MK' = KRI_v \cdot D$ and then recovers the actual message key MK from MK' using KRI_{ECC} . As the basis for this computation, it can be shown that $MK' \cdot C - KRI_{ECC} = (X \cdot E \cdot M_B) \cdot C$, and that $MK' = MK + X \cdot E \cdot M_B$. Thus, Alice is
25 able to derive MK from MK' using the known values of the matrix C and the vector KRI_{ECC} .

Alice may use an error correction computation to derive MK from MK'. For this purpose, it can be shown that the vector elements of $MK' \cdot C - KRI_{ECC}$ can be written
30 as

$$(MK' \cdot C - KRI_{ECC})_i = \sum_{j=1}^r (S_{j,1} \cdot \mu_{j,1} + S_{j,2} \cdot \mu_{j,2} + S_{j,3} \cdot \mu_{j,3}) \cdot C_{j,i},$$

wherein $\mu_{j,1}, \mu_{j,2}, \mu_{j,3}$ are small random variables. The correct values of these random variables for a given KRI can be found by searching through all reasonable values of $\mu_{j,1}, \mu_{j,2}, \mu_{j,3}$ (i.e., with the correct expected value and variance) over the rows i until the correct values of are found. The expected value of $\mu_{j,\alpha}$ is $\frac{h.k_\alpha}{2}$, while the standard deviation of $\mu_{j,\alpha}$ is $\frac{\sqrt{h.k_\alpha}}{2\sqrt{2}}$. In these expressions, h is the number of non-zero elements in each column of the matrix E ; k_α is the number of elements in the j^{th} column of the matrix M_B all having the same coefficient $S_{j,\alpha}$; and α has the value 1, 2 or 3.

When the ECC matrix C is constructed as suggested above (with $l = r - 1$ and only elements $C_{i,i}$ and $C_{i+1,i}$ non-zero), the computation is simplified, since only six of the random variables must be considered for each value of i ($\mu_{1,1}, \mu_{1,2}, \mu_{1,3}, \mu_{2,1}, \mu_{2,2}, \mu_{2,3}$ for $i = 1$, and so forth). Thus, for l sufficiently large, only the single correct solution for the random variables will be upon completion of the ECC computation. This solution is then used in finding MK.

In embodiments of the present invention, values of the parameters n , m and r are chosen so as to ensure the security of the HLV scheme against attack. Typically, n is much larger than $m+r$, for example, $n > 4(m+r)$, and therefore the public key and KRI do not provide enough linear equations to recover X (i.e., the numerical seed has greater entropy than the KRI and than the message key). Typically, $n \geq 200$.

To avoid other kinds of attacks, it is desirable that m be much larger than r , for example, $m > 3r$.

Typically, $m > 40$, and c_1 , c_2 and c_3 are all of the same order. The matrix E is selected carefully to avoid small vectors in the lattice spanned by the vectors in A and in B within the n -dimensional vector space over Z_p .

5 The security of the scheme described above may be enhanced in a number of ways:

1. The scheme may be repeated several (for example, ten) times, and the actual message key is a concatenation of all the recovered keys. This approach is simple but
10 requires ten times more bandwidth and ten times the number of computations.
2. Alternatively, the scheme may be launched only once, but with larger parameters. For example, the B matrix may have ten times more columns ($10 \cdot r$), and likewise
15 the ECC matrix C ($10 \cdot l$). The columns may be divided into 10 groups of l columns each, wherein the first group corresponds to message key elements $1-r$, the second group to elements $r+1 - 2r$, and so forth. The message key is recovered, in the manner described
20 above, over the groups of r elements in succession. To maintain the required entropy relationships and to meet the required security level in this case, the parameters should be chosen so that $m > 20r$.
3. As a further alternative, the message key may be
25 recovered without an explicit error correction step. Instead, the error correction code is integrated into matrix A , which is redefined as a $n \times m+d$ matrix. The extra d columns provide the information needed to record the true values of $y_1 \dots y_m$, as in the original
30 scheme described above.

AUTHENTICATED KEY EXCHANGE USING KEY RECOVERY INFORMATION

Fig. 5 is a flow chart that schematically illustrates a method for secure communication, in accordance with another embodiment of the present invention. This method uses an encryption scheme similar to that described above, but applies it as the basis for an authenticated key exchange between communicating parties, which are again identified as "Alice" and "Bob."

Upon establishing their communication link, Alice and Bob exchange their respective public keys, at a public key exchange step 90. (The private keys and the corresponding public keys are chosen and computed as described above.) Alice and Bob each choose a respective seed, comprising a vector of integers, which may be chosen at random. At a message key generation step, Alice uses her seed together with Bob's public key to generate her message key and KRI; and Bob similarly uses his seed with Alice's public key to generate his message key and KRI. Alice and Bob transmit their respective KRI one to the other, at a KRI exchange step 94.

Alice and Bob use the KRI that they have received together with their own private keys to compute each other's message keys, at a key recovery step 96. In other words, Alice uses her own private key together with Bob's KRI to compute Bob's message key; and Bob uses his own private key together with Alice's KRI to compute Alice's message key. Then Alice and Bob use both Alice's and Bob's message keys together in encrypting and decrypting messages that they transmit to and receive from one another, at a key combination step 98. Typically, Alice and Bob each generate a shared key by combining Alice's and Bob's message keys in a predefined

manner, such as by taking a XOR or other combinatorial function of the two keys.

Other applications of the encryption and decryption schemes described above will be apparent to those skilled in the art after reading the above description and are considered to be within the scope of the present invention. It will thus be appreciated that the embodiments described above are cited by way of example, and that the present invention is not limited to what has been particularly shown and described hereinabove. Rather, the scope of the present invention includes both combinations and subcombinations of the various features described hereinabove, as well as variations and modifications thereof which would occur to persons skilled in the art upon reading the foregoing description and which are not disclosed in the prior art.

CLAIMS

1. A cryptographic method, comprising:

receiving a public key belonging to a message recipient having a private key corresponding to the public key;

selecting a numerical seed, having a first entropy;

using the numerical seed and the public key, generating key recovery information having a second entropy, which is less than the first entropy, and generating a message key having a third entropy, which is less than the first entropy;

encrypting a message using the message key; and

transmitting the encrypted message and the key recovery information to the message recipient, whereby the recipient reconstructs the message key using the key recovery information and the private key and decrypts the encrypted message using the reconstructed message key.

2. The method according to claim 1, wherein generating the key recovery information and generating the message key comprise performing linear algebraic operations.

3. The method according to claim 2, wherein the key recovery information and the message key are generated and the message key is reconstructed without performing exponentiation.

4. The method according to claim 2 or 3, wherein the numerical seed comprises a set of integer values, wherein the public key comprises a matrix of first integer entries and a vector of second integer entries, and wherein generating the message key comprises multiplying the integer values by the vector, and generating the key

recovery information comprises multiplying the integer values by the matrix.

5 5. The method according to claim 4, and comprising generating the public key by iteratively performing modulo operations and multiplications using elements of the private key.

6. The method according to any of claims 2-5, and comprising reconstructing the message key using linear algebraic operations.

10 7. The method according to claim 6, wherein reconstructing the message key comprises iteratively performing modulo operations and multiplications using the key recovery information and elements of the private key.

15 8. The method according to claim 2 or 3, wherein the public key is derived from a set of hidden linear variables, comprising vectors in a finite field, and a set of error vectors in the finite field, and wherein generating the key recovery information comprises
20 computing a key recovery vector and an error correcting code using the public key.

9. The method according to claim 8, and comprising reconstructing the message key using the private key, by performing an error correction computation on the message
25 key using the error correcting code.

10. The method according to any of claims 1-9, wherein encrypting the message comprises applying a symmetric cipher to the message using the message key.

11. The method according to any of claims 1-10, wherein
30 the third entropy is less than the second entropy.

12. The method according to any of claims 1-11, wherein generating the key recovery information and the message key comprises generating first key recovery information and a first message key, and wherein encrypting the message comprises:

receiving second key recovery information from the message recipient;

reconstructing a second message key using the second key recovery information; and

encrypting the message using the first and second message keys together.

13. A cryptographic method, comprising:

holding a private key at a message recipient, while a public key corresponding to the private key is provided to a sender;

receiving from the sender key recovery information computed by the sender using the public key and a selected numerical seed, wherein the numerical seed is not transmitted to the message recipient and has a first entropy, while the key recovery information has a second entropy, which is less than the first entropy;

receiving from the sender a message, which is encrypted using a message key computed by the sender using the integer values and the public key, wherein the message key is not transmitted to the message recipient and has a third entropy, which is less than the first entropy; and

reconstructing the message key at the message recipient using the key recovery information and the private key; and

decrypting the encrypted message using the reconstructed message key.

14. The method according to claim 13, wherein reconstructing the message key comprises applying linear algebraic operations to the key recovery information using elements of the private key.

5 15. The method according to claim 14, wherein the message key is reconstructed without performing exponentiation.

16. The method according to claim 14 or 15, wherein reconstructing the message key comprises iteratively
10 performing modulo operations and multiplications using the key recovery information and the elements of the private key.

17. The method according to claim 14 or 15, wherein the key recovery information comprises an error correcting
15 code, and wherein reconstructing the message key comprises performing an error correction computation using the error correcting code.

18. Cryptographic apparatus, comprising:

a memory, which is configured to hold a public key
20 belonging to a message recipient having a private key corresponding to the public key; and

a processor, which is configured to select a numerical seed having a first entropy, to use the numerical seed and the public key in generating key
25 recovery information having a second entropy, which is less than the first entropy, and in generating a message key having a third entropy, which is less than the first entropy, to encrypt a message using the message key, and to transmit the encrypted message and the key recovery
30 information to the message recipient, whereby the recipient reconstructs the message key using the key

recovery information and the private key and decrypts the encrypted message using the reconstructed message key.

19. The apparatus according to claim 18, wherein the key recovery information and the message key are generated by performing linear algebraic operations.

20. The apparatus according to claim 19, wherein the key recovery information and the message key are generated and the message key is reconstructed without performing exponentiation.

21. The apparatus according to claim 19 or 20, wherein the numerical seed comprises a set of integer values, wherein the public key comprises a matrix of first integer entries and a vector of second integer entries, and wherein the processor is configured to generate the message key by multiplying the integer values by the vector, and to generate the key recovery information by multiplying the integer values by the matrix.

22. The apparatus according to claim 21, wherein the public key is generated by iteratively performing modulo operations and multiplications using elements of the private key.

23. The apparatus according to any of claims 19-22, wherein the message key is reconstructed by the recipient using linear algebraic operations.

24. The apparatus according to claim 23, wherein the message key is reconstructed by iteratively performing modulo operations and multiplications using the key recovery information and elements of the private key.

25. The apparatus according to claim 19 or 20, wherein the public key is derived from a set of hidden linear

variables, comprising vectors in a finite field, and a set of error vectors in the finite field, and wherein the key recovery information comprises a key recovery vector and an error correcting code, which are computed using
5 the public key.

26. The apparatus according to claim 25, wherein the message key is reconstructed by the recipient using the private key, by performing an error correction computation on the message key using the error correcting
10 code.

27. The apparatus according to any of claims 18-26, wherein the processor is configured to encrypt the message by applying a symmetric cipher to the message using the message key.

15 28. The apparatus according to any of claims 18-27, wherein the third entropy is less than the second entropy.

29. The apparatus according to any of claims 18-28, wherein the key recovery information and the message key
20 generated by the processor comprise first key recovery information and a first message key, and wherein the processor is configured to receive second key recovery information from the message recipient, to reconstruct a second message key using the second key recovery
25 information, and to encrypt the message using the first and second message keys together.

30. Cryptographic apparatus, comprising:

a memory, which is configured to hold a private key and a public key corresponding to the private key; and

a processor, which is configured to receive from a sender key recovery information, which is computed by the sender using the public key and a numerical seed, and a message, which is encrypted using a message key computed
5 by the sender using the integer values and the public key,

wherein the numerical seed and the message key are not transmitted by the sender, and wherein the numerical seed has a first entropy, while the key recovery
10 information has a second entropy, which is less than the first entropy, and the message key has a third entropy, which is less than the first entropy, and

wherein the processor is configured to reconstruct the message key using the key recovery information and
15 the private key, and to decrypt the encrypted message using the reconstructed message key.

31. The apparatus according to claim 30, wherein the processor is configured to reconstruct the message key by applying linear algebraic operations to the key recovery
20 information using elements of the private key.

32. The apparatus according to claim 31, wherein the message key is reconstructed without performing exponentiation.

33. The apparatus according to claim 31 or 32, wherein
25 the processor is configured to reconstruct the message key by iteratively performing modulo operations and multiplications using the key recovery information and the elements of the private key.

34. The apparatus according to claim 31 or 32, wherein
30 the key recovery information comprises an error correcting code, and wherein the processor is configured

to reconstruct the message key by performing an error correction computation using the error correcting code.

35. A computer software product, comprising a computer-readable medium in which program instructions are stored, which instructions, when read by a processor, cause the processor to receive a public key belonging to a message recipient having a private key corresponding to the public key, to select a numerical seed having a first entropy, to use the numerical seed and the public key in generating key recovery information having a second entropy, which is less than the first entropy, and in generating a message key having a third entropy, which is less than the first entropy, to encrypt a message using the message key, and to transmit the encrypted message and the key recovery information to the message recipient, whereby the recipient reconstructs the message key using the key recovery information and the private key and decrypts the encrypted message using the reconstructed message key.

36. A computer software product, comprising a computer-readable medium in which program instructions are stored, which instructions, when read by a processor, cause the processor to access a memory holding a private key and a public key corresponding to the private key, to receive from a sender key recovery information, which is computed by the sender using the public key and a numerical seed, and a message, which is encrypted using a message key computed by the sender using the integer values and the public key,

wherein the numerical seed and the message key are not transmitted by the sender, and wherein the numerical

seed has a first entropy, while the key recovery information has a second entropy, which is less than the first entropy, and the message key has a third entropy, which is less than the first entropy, and

5 wherein the instructions cause the processor to reconstruct the message key using the key recovery information and the private key, and to decrypt the encrypted message using the reconstructed message key.

37. A cryptographic method, comprising:

10 receiving a public key belonging to a message recipient having a private key corresponding to the public key;

 selecting a numerical seed;

 using the numerical seed and the public key,
15 generating key recovery information comprising a key recovery vector and an error correcting code, and
 generating a message key having a third entropy, which is less than the first entropy;

 encrypting a message using the message key; and

20 transmitting the encrypted message and the key recovery information to the message recipient, whereby the recipient reconstructs the message key using the key recovery information and the private key and decrypts the encrypted message using the reconstructed message key.

25 38. The method according to claim 37, wherein the public key is derived from a first set of hidden linear variables, comprising vectors in a finite field, and a set of error vectors in the finite field.

 39. The method according to claim 37 or 38, and
30 comprising reconstructing the message key using the private key, by performing an error correction

computation on the message key using the error correcting code.

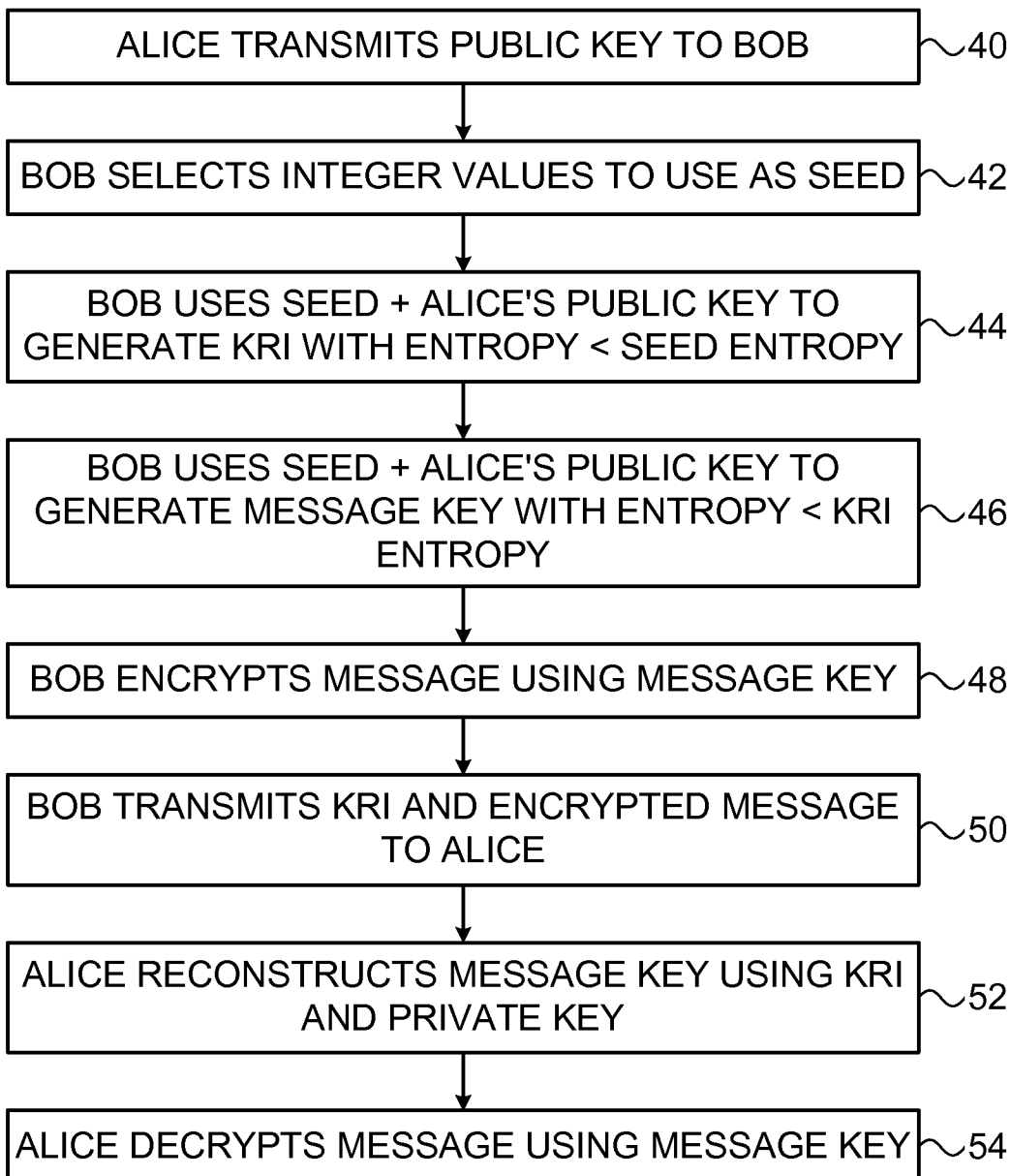
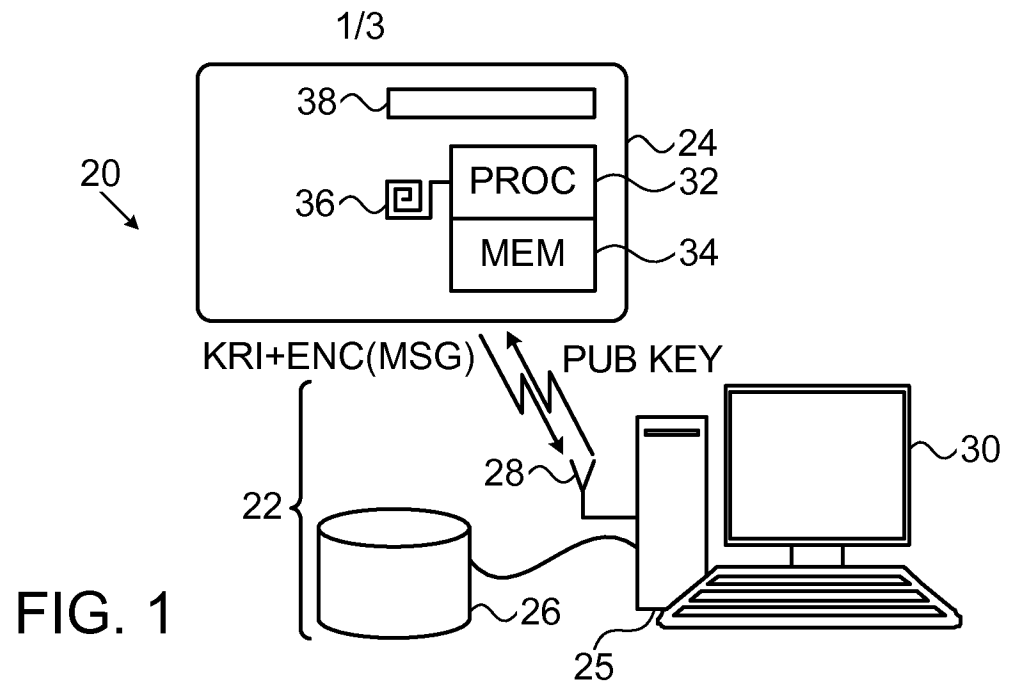
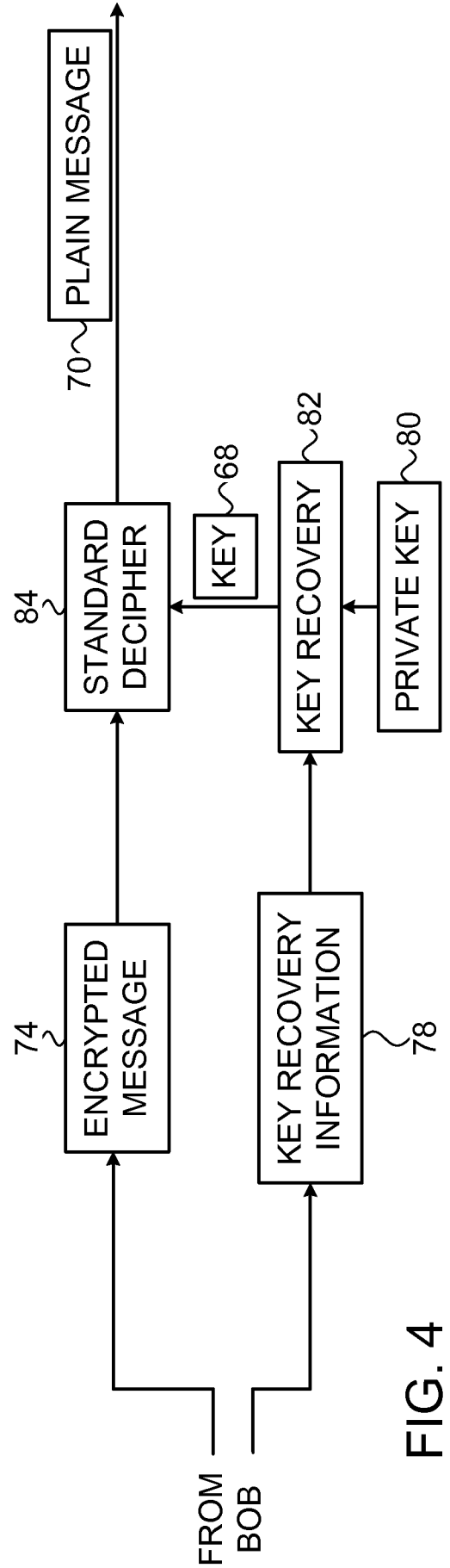
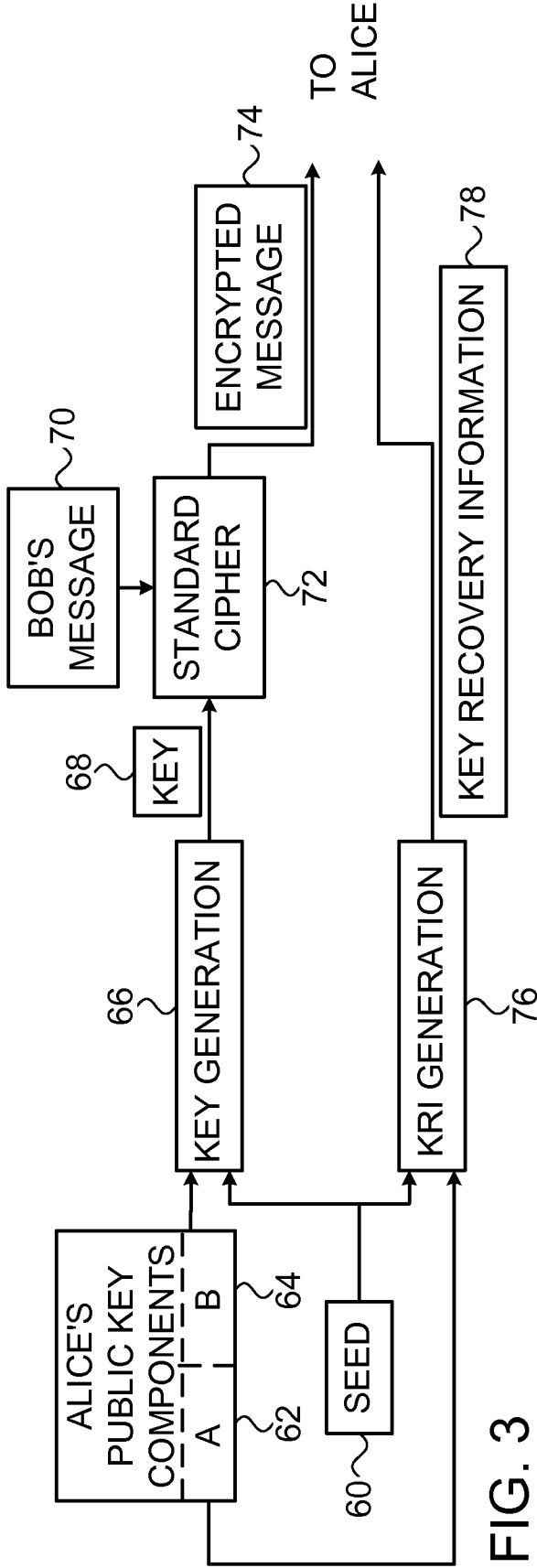


FIG. 2



3/3

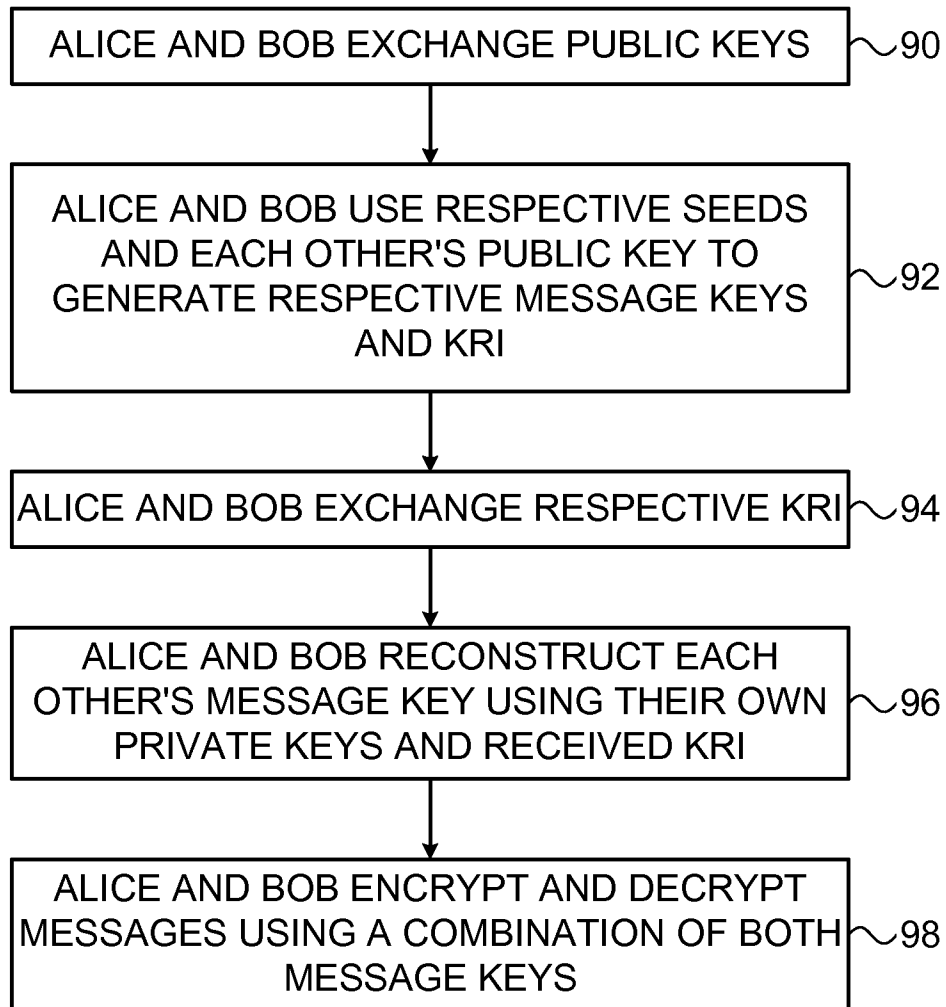


FIG. 5

INTERNATIONAL SEARCH REPORT

International application No

PCT/IB2012/052930

A. CLASSIFICATION OF SUBJECT MATTER

INV. H04L9/30

ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EIICHIRO FUJISAKI ET AL: "Secure Integration of Asymmetric and Symmetric Encryption Schemes", FIELD PROGRAMMABLE LOGIC AND APPLICATION, vol. 1666, 1 January 1999 (1999-01-01), pages 537-554, XP055045327, Berlin, Heidelberg ISSN: 0302-9743, DOI: 10.1007/3-540-48405-1_34 ISBN: 978-3-54-045234-8	1,6,10, 11,13, 14,18, 19,23, 27,28, 30,31, 35,36
Y	section 3; figure 1 ----- -/-	3,8,9, 12, 15-17, 20,25, 26,29, 32-34



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

23 November 2012

Date of mailing of the international search report

03/12/2012

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

Billet, Olivier

INTERNATIONAL SEARCH REPORT

International application No

PCT/IB2012/052930

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	"Chapter 12: Key Establishment Protocols ED - Menezes A J; Van Oorschot P C; Vanstone S A", HANDBOOK OF APPLIED CRYPTOGRAPHY; [CRC PRESS SERIES ON DISCRETE MATHEMATICS AND ITS APPLICATIONS], CRC PRESS, BOCA RATON, FL, US, PAGE(S) 489 - 541 1 October 1996 (1996-10-01), XP001525012, ISBN: 978-0-8493-8523-0 Retrieved from the Internet: URL:http://www.cacr.math.uwaterloo.ca/hac/ section 12.15	1,6, 10-14, 18,19, 23, 27-31, 35,36
Y		3,8,9, 12,15, 16,20, 25,26, 29,32-34
Y	----- JINTAI DING ED - FENG BAO ET AL: "A New Variant of the Matsumoto-Imai Cryptosystem through Perturbation", 26 February 2004 (2004-02-26), PUBLIC KEY CRYPTOGRAPHY & PKC 2004; [LECTURE NOTES IN COMPUTER SCIENCE;;LNCS], SPRINGER-VERLAG, BERLIN/HEIDELBERG, PAGE(S) 305 - 318, XP019002844, ISBN: 978-3-540-21018-4 sections 1 and 2 -----	3,8,9, 15-17, 20,25, 26,32-34
A	"Chapter 8: Public-Key Encryption ED - Menezes A J; Van Oorschot P C; Vanstone S A", HANDBOOK OF APPLIED CRYPTOGRAPHY; [CRC PRESS SERIES ON DISCRETE MATHEMATICS AND ITS APPLICATIONS], CRC PRESS, BOCA RATON, FL, US, PAGE(S) 283 - 319 1 October 1996 (1996-10-01), XP001525008, ISBN: 978-0-8493-8523-0 Retrieved from the Internet: URL:http://www.cacr.math.uwaterloo.ca/hac/ sections 8.5 and 8.6 -----	1-3,6, 8-20,23, 25-36