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

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
**15 February 2007 (15.02.2007)**

**PCT**

(10) **International Publication Number**
**WO 2007/018476 A1**

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

(21) **International Application Number:** PCT/SG2006/000229

(22) **International Filing Date:** 11 August 2006 (11.08.2006)

(25) **Filing Language:** English

(26) **Publication Language:** English

(30) **Priority Data:**
PI 20053739 11 August 2005 (11.08.2005) MY

(71) **Applicants** *(for all designated States except US)*: **NSS MSC SDN BHD** [MY/MY]; Suite E-07-21, Plaza Mont Kiara, No. 2, Jalan Kiara, Mont Kiara, 50480 Kuala Lumpur (MY). **PINTAS PTE LTD** [SG/SG]; No. 151 Chin Swee Road, #09-11/13, Manhattan House, Singapore (SG).

(72) **Inventors; and**

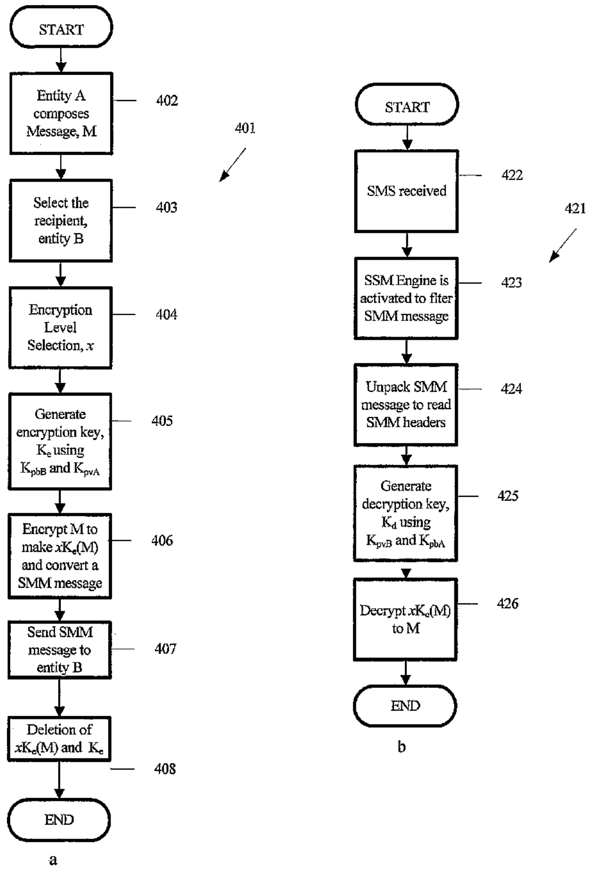
(75) **Inventors/Applicants** *(for US only)*: **MANICKAM, Setha, S.A.** [IN/IN]; NSS India PVT. LTD, 19-A Vidyut Nagar, Koregoan Park, Pune (IN). **SAREEN, Vikram** [MY/MY]; NSS MSC SDN BHD, Suite E-07-21, Plaza Mont Kiara, No. 2, Jalan Kiara, Mont Kiara, 50480 Kuala Lumpur (MY). **NISHAT, Anthony, Navin** [MY/MY]; NSS MSC SDN BHD, Suite E-07-21, Plaza Mont Kiara, No. 2, Jalan Kiara, Mont Kiara, 50480 Kuala Lumpur (MY). **GANGULY, Tushar** [MY/MY]; NSS MSC SDN BHD, Suite E-07-21, Plaza Mont Kiara, No. 2, Jalan Kiara, Mont Kiara, 50480 Kuala Lumpur (MY). **DOGRA, Vijay** [MY/MY]; NSS MSC SDN BHD, Suite E-07-21, Plaza Mont Kiara, No. 2, Jalan Kiara, Mont Kiara, 50480 Kuala Lumpur (MY).

(74) **Agent: CHOW WENG WENG**; No. 151, Chin Swee Road, #09-11/13, Manhattan House, Singapore (SG).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA,

*[Continued on next page]*

(54) **Title:** HYBRID CRYPTOGRAPHIC APPROACH TO MOBILE MESSAGING


START
Entity A composes Message, M
402
401
Select the recipient, entity B
403
Encryption Level Selection, x
404
Generate encryption key, K_e using K_pbB and K_pvA
405
Encrypt M to make xK_e(M) and convert a SMM message
406
Send SMM message to entity B
407
Deletion of xK_e(M) and K_e
408
END
a
START
SMS received
422
421
SSM Engine is activated to filter SMM message
423
Unpack SMM message to read SMM headers
424
Generate decryption key, K_d using K_pvB and K_pbA
425
Decrypt xK_e(M) to M
426
END
b

(57) **Abstract:** Secure Mobile Messaging via secure, trusted and secure, or trusted message is disclosed. Sender uses recipient's public key and own private key to generate encryption key Ke to encrypt message M with selected encryption level x. The encrypted message xKe(M) is then transmitted over digital network that provide SMS to the recipient. The recipient then uses own private key and sender's public key to generate decryption key Kd (Ke=Kd). According to an embodiment the encryption of SMM messages is performed according to a proprietary cryptographic algorithm. Also disclosed is digital signature generation using sender's private key and a hash result of message M, and DS is then appended to M. The signed message is then transmitted over the digital network. The recipient then uses the sender's public key to generate fresh hash results and authenticates sender's signature. When authenticated the message is shown.

NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SY, 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, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Published:**

— *with international search report*

# Hybrid Cryptographic Approach to Mobile Messaging

## Field of Invention

This invention generally relates to exchange of Secure Mobile Messaging (SMM) over digital telecommunication network using hybrid cryptographic (symmetric and asymmetric keys encryption/decryption) approach applied to the existing mobile messaging protocols but not limited to Short Message Service (SMS), Multimedia Messaging Service (MMS), and mobile-based Email messages.

## Background of the Invention

Short message service (SMS) was introduced in the Global Services for Mobile (GSM) system and was later supported by most of the digital telecommunication networks such as Code Division Multiple Access (CDMA), Time Division Multiple Access (TDMA), and Personnel Communication Services (PCS). A typical SMS message, with a maximum of 70 - 160 characters, can be composed, and sent by a sender's telecommunication device, which can be a mobile device, and transmitted over digital telecommunication network to one or more recipients' telecommunication devices. Although SMS is intended for use with mobile telecommunication devices, it can nonetheless be sent from or received by other devices in a telecommunication network such as a landline telephone, a personal digital assistant (PDA), a palmtop, a computer, and a gateway server (which can be an SMS gateway or any other application server). The sender and receiver of such mobile messages need not only a person, but can be a mobile itself on automated response mode, a computer, a PDA, and a gateway server.

The transmitted SMS message can be intercepted at many stages in the digital mobile communication network. Among those, four prominent stages of interception are described as illustrated purpose. Firstly, the messages can be intercepted during its route from a sender's mobile telecommunication device to the base station, i.e. over the air (OTA). Secondly, it can be between the base station and SMS Center. Thirdly, it may be between one Mobile Switching Center (MSC) to another MSC. Lastly, it may be intercepted between the base station and recipient's telecommunication device in OTA. Existing SMS and its protocol as defined by current digital telecommunication network specifications is insecure and not 2 suitable for transmitting private, confidential, privileged or sensitive information through existing network available. Present SMS protocol also does not provide any means for authenticating an SMS sender and validating its content integrity. As such, SMS messages can be repudiated and cannot be further applied in many other applications (like bank, financial, stock transactions etc) until a means for authenticating and validating SMS message integrity is provided. The same vulnerability is also present in other messaging standards in mobile communication like MMS and email on mobile device.

The new system proposes the creation of a secure and trusted messaging system without the introduction of new standards to mobile digital network system. It can ride on current mobile communication protocols such as mobile originated (MO) or mobile terminated (MT) SMS, MMS, and email.

Currently, symmetric encryption approach has been utilized to send secure MO and secure MT SMS messages. Although symmetric encryption requires lesser resources for execution of a prescribed encryption level and is faster, it is a known fact that symmetric encryption approach has key management problems. This is because the encryption key has to be transmitted over insecure channels to the recipient for decrypting the message.

Whereas the asymmetric encryption system requires high computing resources and is very slow for encryption and decryption, but it is quite good for key management due to its use of public and private key pair. A first entity's public key, which is used by a second entity only to encrypt a message intended for the former, can be shared with all those who want to send a secure message to the first entity that uses its private key and a sender's public key to decrypt messages. Using the best of both symmetric and asymmetric encryption systems, a hybrid system is used, in which the asymmetric system is used for key management and the symmetric system is used for encryption and decryption of messages.

Asymmetric key algorithms such as Rivest Shamir Adleman (RSA), Diffie-Helmen (DH), and Elliptic Curve Cryptography (ECC) algorithms require more computing resources than symmetric encryption algorithm for a given key length. However, the ECC algorithm with less key sizes has an equal strength as compared to RSA algorithm with large key sizes. For example, the strength of 163 bits of key size of ECC is equivalent of 1024 bits of key size of RSA algorithm. Implementing such high resources consumption algorithms have made it a challenge for MO or MT secure messaging on an embedded mobile system environment.

## Summary of Invention

The objective of the invention is to provide methods for sending and receiving secure MT or MO messages in existing digital telecommunication network using a hybrid cryptographic approach without introduction of new standards.

It is a further objective of the invention to provide an encryption of a SMS or MMS or email message in one block with small arrays using a novel 32-bit base proprietary symmetric encryption algorithm in place of existing symmetric algorithm. This saves a 3 to 12 characters to increase the maximum allowable text size in an SMM enabled messages.

The invention is the application of hybrid cryptographic approach, which includes encryption, decryption, signing, and verification processes, to a mobile telecommunication device or any telecommunication device that can send a mobile terminated (MT) or mobile originated (MO) messages securely, also called Secure Mobile Messaging (SMM) messages, via a digital telecommunication network. In this specification, “telecommunication device” refers generally to a mobile messaging enabled device that can send or receive mobile messages. This doesn’t necessarily means that device must mobile, but the messages can always be received and sent from a mobile device which is characterized by limited processing resources. This hybrid cryptographic approach has been practically realized in a modified application on board a mobile telecommunication device as disclosed in another co-pending Malaysia Patent Application by the applicant.

In this invention, asymmetric approach to key management is adopted wherein a registered user, a landline telephone, a PDA, a palmtop, a computer, a gateway server (which can be an SMS gateway or any other application server), or any mobile messaging enabled telecommunication device on automated response mode, which throughout the specification is referred to as an entity, can generate its own public and private key pair during installing and activating of the application through license process. An encryption key, $K_e$ is then generated using the entity's private key and an intended recipient's public key. The entity, in this case being the sender of a MO message, possesses beforehand the intended recipient's public key by a means of manual or automated key exchange mechanisms provided in the key management means.

In the hybrid approach to encryption/decryption of a message, the encryption key, $K_e$ and decryption key, $K_{d,}$ are symmetrical to each other (i.e., $K_e = K_d$). The messages may be encrypted and decrypted using one of the existing symmetrical encryption algorithms used together with the encryption key, $K_e$ and decryption key, $K_d$. The invention may also optionally use a novel proprietary symmetric encryption algorithm that allows a higher maximum allowable text size in encrypted SMM messages.

Furthermore according to this invention, the entity not only has the option to "encrypt only" but also to "sign only" or "encrypt and sign" to yield a trusted message or a secure and trusted message for exchange over digital telecommunication network. This is achieved by encapsulating the trusted message or the secure and trusted message, in the embodiment of this invention into a SMM mobile message body. An asymmetric digital signature approach is also adopted to sign and verify the MO and MT message respectively to accommodate trusted messaging service.

The invention may utilize a key management means and method for transmitting, receiving and storing identification information of a plurality of registered entities. The key management means comprises at least a computer server and related system elements that are capable of communicating with a mobile device via digital communication network. The key management means allows a group of users of this system to manually or automatically exchange the keys "on the fly" without the users' intervention. User identification information that is stored on the computer server includes telephone number and associated public key, which are tightly bound and coupled to each other. In absence of the explained key management means, the invention also provides a means for peer-to-peer keys exchange to its users.

The invention is a hybrid cryptographic approach that is suitably modified to be easily executed on lower computing resources. In the light of this, the invention that is a hybrid approach to encryption/decryption of an MT or MO messages further comprising an asymmetric approach to signing and verifying the MT or MO messages that is not only limited to mobile communication terminal, but can also be applied to other communication elements in a communication network (as exemplified by a server to be further disclosed later). A server that can send and receive SMM mobile messages is referred to as an "Enterprise Server" (ES) herein and throughout this specification. It is also conceivable that as the need arises the invention, on its entirety or rather in part, can be modified and adapted for use in other embedded systems which main purpose may not be secure communications but involved some form of communications in its execution.

## Brief Description of the Drawings

The drawings constitute a part of this specification and include exemplary embodiments to the invention, which may be embodied in various forms. It is to be understood that in some instances various aspects of the invention may be extricated from the entire invention and shown alone by itself to facilitate an understanding of the invention.

Figure 1a illustrates a representation of a SMS protocol according to the GSM standards (prior art), comprising of a SMS Identification Header (101) and a SMS Text Body (102). Figure 1b illustrates a representation of the composition of a Secure Mobile Messaging (SMM) with its protocol.

Figure 2 is a table showing size of composable text body for different messages that are encrypted only, signed and encrypted, and signed only for respective SMS text, MMS, and Unicode representation for AES and proprietary cryptographic algorithms.

Figure 3a is a flow diagram illustrating process steps of entity A and entity B generating their public and private key pairs and exchanging their public keys with each other according to the embodiment of this invention.

Figure 3b is a flow diagram illustrating process steps of entity A requesting entity B's public key, $K_{pbB}$, from entity B and storing the key after receiving it from entity B according to the embodiment of this invention.

Figure 3c is flow diagram illustrating the process steps of storing an entity's public key in a key repository of the enterprise server (ES) according to the embodiment of this invention.

Figure 3d is a flow diagram illustrating process steps of entity A requesting entity B's public key, $K_{pbB}$, from Enterprise Server (ES) according to the embodiment of this invention.

Figure 3e is flow diagram illustrating the process steps of server generating its public and private key and uploading the public key in the repository according to the embodiment of this invention.

Figure 3f is flow diagram illustrating the process steps of server public key given to a new entity whenever the latter is activated according to the embodiment of this invention.

Figure 3g is flow diagram illustrating the process steps of server public key broadcasting to all of its subscribers whenever there is a renewal or change of server's key pair according to the embodiment of this invention.

Figure 4a is a flow diagram illustrating process steps of entity A encrypting a message to be sent to entity B according to the embodiment of this invention.

Figure 4b is a flow diagram illustrating process steps of entity B decrypting an encrypted message received from entity A according to the embodiment of this invention.

Figure 5a is a flow diagram illustrating process steps of entity A digitally signing and encrypting a message to be sent to entity B according to the embodiment of this invention.

Figure 5b is a flow diagram illustrating process steps of entity B decrypting and verifying a digitally signed and encrypted message received from entity A according to the embodiment of this invention.

Figure 6a is a flow diagram illustrating process steps of entity A digitally signing a message to be sent to entity B according to the embodiment of this invention.

Figure 6b is a flow diagram illustrating process steps of entity B verifying a digitally signed message received from entity A according to the embodiment of this invention.

## Detailed Description of the Invention

Detailed descriptions of the preferred embodiment which summarize all aspects of different embodiments of this invention are provided herein and it is appreciated that these aspects may be implemented individually or in any combination. It is also understood that the present invention may be embodied in various forms. Therefore, specific details disclosed herein are not to be interpreted as limiting, but rather as a basis for the claims and as a representative basis for teaching one skilled in the art to employ the present invention in virtually any appropriately detailed system, structure or manner.

The method for sending secure, trusted and secure, and trusted messages disclosed in the present invention, which is also referred to as a Secure Mobile Messaging (SMM) system, uses conventional messaging protocol via digital telecommunication network. In particular, the invention disclosed herein is in relation to its application to SMS messages. It is conceivable and apparent to a person skilled in the art that the invention may be applied for other mobile messages such as MMS messages and mobile-based emails by modifying the embodiment of the application for carrying out the method disclosed herein. It is also further conjectured that invention could be applied to digital communication function of any embedded system.

Java 2 Platform Micro Edition (J2ME), Windows CE, and Symbian are existing operating systems used for writing applications for mobile phones and can be used for writing a component or a engine and applications for carrying out this invention. It is also conceivable that any embedded OS developed in the future can be used for writing similar components or engines and related applications for executing this invention.

Another embodiment of this invention is to apply a public key based hybrid cryptographic approach for securing Mobile messaging applications like SMS, MMS, and Email, which rides on existing mobile communication protocol without any modification. This novel concept is different from securing the classical Wireless Application Protocol (WAP) application using WAP-public key infrastructure (PKI) techniques. The WAP, which uses General Packet Radio Service (GPRS) standard for faster data transmission, is a mobile Internet technology that allows mobile users to browse the Internet using WAP-enabled browsers.

One embodiment of the invention is exemplified by an engine or component that will convert mobile messages (SMS, MMS, and Email) to be SMM messages. Other applications are associated with the engine to carry out various possible tasks which include key pair generation, public key exchange, public key request and storing, message encryption, decryption, signing, and verifying using the generated key pair. It is possible that, apart from secure mobile-to-mobile communication, similar engine and associated application for working on converted SMM messages can also be implemented on a server to enable mobile-to-computer and mobile-to-server messaging and vice-versa which will be further disclosed later.

The invention which is also referred to as a Secure Mobile Message (SMM) system, provides additional features such as sending secure, trusted and secure, and trusted messages as a mobile message. According to the one of the preferred embodiment of the invention, the additional features are all implemented by utilizing an engine and application software as part of the solution, both of which integrate seamlessly with application layer of existing messaging application on smart mobile device. It is reiterated that the existing messaging application may be a SMS, a MMS or a mobile email application. Thus neither modification on the smart mobile devices nor changing the underlying messaging format of SMS, MMS or email is needed to achieve a secure mobile messaging.

The following discloses different aspects of this invention and it is appreciated that these aspects may be implemented individually or in any combination.

Figure 1a illustrates the prior art mobile messaging protocol format that contains a mobile messaging header (101) and a text body (102) as per the GSM standards. The header includes fields identifying the mobile messaging type, parameters identifying the mobile service provider, source and destination addresses, and other fields as specified in GSM standards. Figure 1b illustrates a SMM message protocol format that retains the existing mobile messaging header (101) that allows it to be fully compatible with the conventional mobile messaging protocol. However the mobile messaging text body (102) is modified so that it becomes a SMM text body (202) which includes additional information, namely the SMM protocol to enable the execution of the invention.

The SMM identification label (210) and SMM type label (211) are contained in an SMM header (209) represented in Figure 1b, which constitutes the SMM protocol. Depending on whether a message is encrypted only, signed and encrypted, and signed only, additional information such as encryption algorithm indicator, encryption level, signature indicator and hash length label are encapsulated in the SMM type label (211). If the SMM message is sent from or sent to a server, the additional information may also include a destination server user's identity (UID), which is included in the SMM type label (211).

The aforementioned information allows an engine, which is a different invention disclosed in a co-pending Malaysia Patent Application by the applicant, installed in the mobile device to recognize whether the message is an encrypted only message, an encrypted and signed message, or signed only message. Digital Signature (213) is an optional information depending on the signing option that is included as part of the SMM text body. Depending on the encryption option, the text (212) and digital signature (213), if present, are encrypted and makes up part of the message body. Otherwise, the rest of the modified message body would contain the message itself. In order to further ensure that the converted SMS message (which is now a SMM message) can travel through different telecommunication networks, the SMM type label (211), the optional Digital signature (213) and the encrypted message (212) are encoded and the encoding factor (214) constitutes the increase in the size of SMM text body.

The length of text composed in SMM message may range from 16 to 112 bytes depends on the options mentioned and various types of text representation such as plain text, multimedia, and Unicode representation adopted. Figure 2 shows the maximum text body sizes available for message composition under different options of encryption, signing and encryption, and signing only, and text encoding using AES and a new proprietary symmetric encryption algorithm in this embodiment of the invention. As per embodiment of this invention, the use of the proprietary cryptographic algorithm increases 3 to 12 text characters for a SMM text body when compared to the use of AES algorithm.

The invention can be implemented according to the following scenario. Each entity, as defined earlier, may first register with an enterprise server (ES) that issues the SMM service. The enterprise server verifies the identity of each of the entity intending to use the SMM service according to predefined procedures prescribed in published standard. Information of each verified entity such as name, address, personal identification number (PIN), personal password and other security data will be kept in a secure manner in the ES database. At registration, ES issues a SMM license to the registered entity. A key pair consisting of a public key and a private key will be generated in the entity's device (e.g., smart mobile, PDA, computer or server) having the issued SMM capabilities as shown in step (300a) and (300b) in Figure 3a. The Elliptic Curve Cryptography (ECC) approach is the preferred and suitable asymmetric cryptographic method for the embedded devices for generating an entity's public key and private key. A copy of both public and a protected private key (if required) may then be transmitted to be stored at the ES along with the entity database in a secure manner.

According to the same scenario, entity A, which can be a mobile device user, the ES or other telecommunication devices can communicate with entity B, which can be another mobile device user, the ES or other telecommunication device in a secure, trusted and secure or trusted manner. All telecommunication devices must be similarly equipped with the means for generating their own private and public key pairs according to step (300a or 300b) shown in Figure 3a and the means for encrypting, decrypting and the means for signing and verifying a SMM message using their key pairs.

Public keys of all entities, regardless of the type of telecommunication devices used, need to be exchanged with one another (301a) & (301b). One entity can provide its public key to another entity by exchanging public key directly (310) as shown in Figure 3a or through the ES indirectly, which is discussed later. In other words, the public key of all entities can be distributed in entity-to-entity manner which covers a mobile-to-mobile scenario, a mobile-to-

ES scenario or scenario that involves any two SMM enabled mobile messaging devices. "Mobile" refers to SMM capable mobile device, "ES" refers to SMM issuing server and "SMM enabled mobile messaging devices" refers to other any other types of telecommunication devices such as PDA, palmtop, and computers herein and throughout the specification. Mobile users, enterprise server and all other telecommunication devices generate their own private and public key pairs (300a or 300b) as shown in Figure 3a to allow them to communicate with each other e.g. entity A and entity B according to the method taught by this invention. Preferably all key pairs are generated according to ECC approach.

Figure 3b flow diagram illustrates how one entity requests another entity's public key (301). Supposedly two entities, designated for purpose of description as entity A and entity B are the registered entities of the SMM service. When entity A wishes to send a secure, trusted and to entity B. Upon receiving the request (304), the inbox handler in entity B's SMM engine will authenticate Msecure, or trusted message to entity B for the first time, entity A has to request entity B's public key, $K_{pbB}$. According to steps (302, 303), entity A sends a public key requisition message, $M_{kB}$ $_{kB}$ status (305a) which includes checking whether requisition message $M_{kB}$ is valid or has not been tampered with. The public key requisition process 301 will be terminated if the authentication of the requisition message, $M_{kB}$ fails (306). If the requisition message, $M_{kB}$ is authenticated, then entity B will be prompted about the requisition (305b) and entity B has the option of either to send (308) or to deny (307) his public key, $K_{pbB}$ to entity A. For key exchange to take place, entity B should send his public key to entity A and the latter will store the entity B's public key, $K_{pbB}$ in its own SMM enabled contacts book (309). Similarly, entity B can request for entity A's public key according to process 301. Entity A and entity B have exchanged their public keys as shown in Figure 3a (310) with each other when both entity A and entity B have obtained each other's key through process 301. The public key exchange (310) also can take place by one of the entities taking the initiative to send his/her public key to another intended entity. This is done by composing a SMM message which contains the entity's own public key and sending it out to the intended entity.

Figure 3c illustrates the process 311 of storing an entity's public key in a key repository of the ES so that other entities can request for it from the ES later. The flow starts with entity B who composes (312) and sends a public key storing request (313) $M_{KpbB}$ to the SMM ES for storing its own public key $K_{pbB}$. Upon receipt, the storing message $M_{KpbB}$ is handled by the inbox handler of the ES (314). The ES will check the validity of the entity's B number and entity's B subscription license (315). If entity B's number and subscription license do not pass the check, the ES will generate a status report (316) and sends it to entity B. If the check (315) is successful, the inbox handler checks if $K_{pbB}$ is already present in the key repository (317). If $K_{pbB}$ is not present in the repository, it will be stored in the repository (319). On the other hand, if $K_{pbB}$ exists in the repository, inbox handler will prompt the entity B whether it intends to overwrite it (318). If entity B chooses to overwrite it, $K_{pbB}$ will be stored in the repository (319). If the entity chooses not to overwrite it, the whole workflow is ended.

Now, entity A can request entity B's public key, $K_{pbB}$ according to the process illustrated in Figure 3d from ES (321) in a mobile-to-ES scenario for sending secure, trusted and secure, or trusted messages. Entity A sends a requisition message, $M_{kB}$, (322, 323) to obtain entity B's public key, $K_{pbB}$, to ES. A number of ways can be used to authenticate entity A, one of which is to require entity A to send his public key in the requisition message to be verified against a copy of the same stored in the ES. Upon receiving the request (324) the inbox handler in the ES will authenticate entity A's number and license status (325a). If the authentication of entity A and permission to request for entity B's public key fails, outbox handler of ES will generate a status report (326) and send to entity A. If entity A is otherwise authenticated, ES will proceed to authenticate entity B's number and license status (325b). If entity B's accessibility and availability fails, outbox handler of ES will generate a status report (327) and send to entity A. If entity B status is otherwise valid, ES will then request entity B's public key, $K_{pbB}$, from the key repository (328) and outbox handler of ES will generate a message (329) which includes $K_{pbB}$ and send to entity A. Upon receiving the message from ES, entity A saves $K_{pbB}$ in its SMM contacts book (330). It must be noted that public keys of entities other than mobile user can also be requested from ES similar to public key request shown in Figure 3d since these public keys are also stored in the ES according to public key storing request shown in Figure 3c.

For exchanging SMM between ES with other entities such as mobile users, the entities need to obtain the server's public key. For this purpose, as with other mobile devices that functions according to this invention, processes (331) of generation of ES key pair and uploading of ES public key are illustrated in Figure 3e. During activation of the ES with server license, the key pair for the ES (332) will be generated. Then the public key of the ES will be uploaded to the repository (334) so that its key can be distributed to all the users of the system as per the invention.

Figure 3f illustrates the process 341 of issuing ES public key to all new entities as per the embodiment of this invention. Whenever a new entity is activated and has generated its key pair, it can upload its public key in the repository as in the process 311. Whenever it is uploaded (342) in the repository, the ES then performs audit check on the record (344) in its repository. If the record does not pass through audit check (344), the initiation of ES' public key dispatch is ended. If the valid record passed through the audit check (344), then ES's public key will be sent (346) or dispatched to every new entity added to the system.

Figure 3g is the workflow process 351 of public key broadcasting by the server as per the embodiment of this invention. The ES may have to regenerate its new key pair (352), whenever there is revocation of these server's keys at anytime after initial activation of the server at the very beginning. The server also has to regenerate its new key pair, whenever the server's private key is lost, corrupted, compromised, or expired. During this time, the server has to broadcast its public key to all its users. After generating its new key pair according to set parameters, it uploads its key to the repository (354). Whenever it is uploaded, the ES reads all the entry in the repository (356) and performs audit checking (358) of each valid record (358) in its repository. If the valid record for a particular entity does not pass through audit check (358), its key will not be dispatched and ES starts reading the next entry in the repository. If the valid record passes the audit check (358), then ES's public key will be sent (359) to every valid entry in the repository.

Once registered users and other entities have generated their own private keys and public keys and have exchanged their public keys with other entities as disclosed earlier, every entity can communicate with each other using secure, trusted and secure, or trusted messages. When one of the entities, e.g. entity B receives a SMM message from entity A, the SMM message may be encrypted only, signed and encrypted, or signed only according to encryption (400), signing and encryption (500), and signing (600) process to be outlined later. Then the SMM message has to be decrypted (421), decrypted and verified (521), or verified (621) accordingly so that entity B can read the message M.

Figure 4a is a flow diagram illustrating process steps of sending secure message (401) according to the embodiment of this invention. As an illustrative example, only encryption process 401 is involved in exchanging SMS messages. In other words, entity A desires to send a secure SMM message to entity B. Entity A must first have entity B's public key, $K_{pbB}$ loaded into its mobile telecommunication device either according to process 300 or 321 as outlined earlier. Entity A will compose the message, M to be sent (402) and choose a recipient (403) such as entity B. Entity A then selects the encryption level $x$ (404) and the application residing on entity A's mobile device will generate encryption key, $K_e$, by Elliptic Curve Cryptography, using entity B's public key, $K_{pbB}$, and its own private key, $K_{pvA}$ (405). M is then encrypted (406) by the same application using either a novel proprietary encryption algorithm or Advance Encryption Standard (AES) according to security level $x$ and $K_e$ to produce an encrypted message $xK_e(M)$ containing SMM message type indicator and cryptographic information according to SMM protocol format. After encoding of SMM type (M) and Kwill be destroyed (408) once the SMM message is sent. label and $xK_e(M)$, SMM message is formed according to (202) and is then sent to entity B (407). Then $xK_{ee}$

Figure 4b is a flow diagram illustrating process steps of decrypting a secure message (421) according to the embodiment of this invention. As an illustrative example, entity B receives a SMM message from entity A and only decryption process is involved. When entity B receives a SMS message (422) (which may or may not be a SMM message), the engine is activated to filter, classify and identify SMM message (423) according to the information in the SMM header (210). After decoding the message, the engine then reads the additional information (424) encapsulated in the SMM message body (211) to determine the encryption algorithm, signature information, hash length, encryption level, and server user identity encapsulated in the message. Based on the information, the application generates decryption key, $K_d$ (i.e., $K_d$=$K_e$) using its private key, $K_{pvB}$, and entity A's public key, $K_{pbA}$ (425) and decrypts $x$Ke(M) to M (426).

Figure 5a is a flow diagram illustrating process steps of sending secure and trusted messages (501) according to the embodiment of this invention. As an illustrative example, entity A desires to send a secure and trusted SMM message, $x$$K_e$(DS|M) to entity B where signing processes (505, 506) and encryption processes (504, 507, 508) are involved. In this invention, Elliptic Encryption Digital Signature Approach (ECDSA) is the preferred approach is to sign and verify the message. Entity A must first have entity B's public key, $K_{pbB}$, loaded as outlined earlier. Entity A firstly composes a message, M (502). Then entity A generates a digital signature DS using $K_{pvA}$, and hash result of M (505) and appends DS to M (506). The application generates encryption key, $K_e$, using entity B's public key, $K_{pbB}$, and entity A own private key, $K_{pvA}$ (507) and then encrypts the original text M and DS by the novel proprietary encryption algorithm which is 32-bit base or Advance Encryption Standard (AES) according to the selected encryption level $x$ to produce $x$$K_e$(DS| M) (508). Then a SMM message is formed using signed and encrypted message, SMM message type and identification, and encoding appropriate fields according to SMM protocol format and is sent to entity B (509). Encryption key, $K_e$ and digital signature, DS will be destroyed (510) once the SMM message is sent.

Figure 5b is a flow diagram illustrating process steps of decrypting and verifying a secure and trusted message (521) according to the embodiment of this invention. As an illustrative example when entity B receives a SMM message from entity A, decryption and digital signature verification processes are needed for that message. When entity B receives SMS messages (522), SMM engine is activated to filter, classify and identify SMM message (523) according to the information in the SMM header (210) that is encapsulated in the SMM message body. The application then decodes the message and reads the additional information (211) encapsulated in the SMM message body (524) to determine the encryption algorithm, signature information, hash length, encryption level, and server user identity. Based on this information, the application in entity B generates a decryption key, $K_d$ (i.e., $K_e= K_d$) using $K_{pvB}$ and $K_{pbA}$ (525) and decrypts $x$$K_e$(DS|M) to DS and M separately (526) with the generated decryption key. The application then generates fresh hash result from message M (527) and continues to generate an output value R using the fresh hash result, entity A's digital signature (DS), and entity A's public key. The output value R is then compared with the original value r that is contained in the DS appended to the message. Once R is verified to be the same as the original value r according to step (528), the application will show message M (529) and entity A is verified. Otherwise, message verification error will be shown if sender (in this case entity A) is not verified (530).

Figure 6a is a flow diagram illustrating process steps of sending a trusted message (601) according to the embodiment of this invention. As an illustrative example, only signing process is involved in the case where entity A desires to send a trusted message, M, to entity B. Entity A must first have the entity B's public key, $K_{pbB}$, loaded as outlined earlier. As with earlier cases, entity A firstly composes a Message, M (602) and selects the recipient, entity B from a list (603). The application then generates the digital signature of the message, DS, using $K_{pvA}$ and hash result of M (604) and appends DS to M to produce a trusted message (DS|M) (605). The signed message DS along with additional information according to SMM protocol format are encoded and a SMM message is composed. Then the SMM message is sent to entity B (606). DS will be wiped out (607) once the SMM message is sent. In cases where messages are signed (500, 600), entity A has the option to append his own public key, $K_{pbA}$ to the digital signature and the sent message (encrypted or unencrypted) to allow entity B to authenticate the sender at a later stage.

Figure 6b is a flow diagram illustrating process steps of verifying a trusted message (621) according to the embodiment of this invention. As an illustrative example where entity B receives a signed SMM messages, digital signature verification process is involved. When entity B receives a SMS message (622), engine is activated to filter, classify and identify

SMM message (623) according to the information in the SMM identification header (210) that is encapsulated in the SMM message body. The application then decodes the SMM message and reads the SMM type header (624) in the message to determine signature information, hash length, and server user identity. Then the application generates an output value R (625) using fresh hash of message M and entity A's public key. The output value R is then compared with the original value r which is contained in the DS (626). Once verified according to step (626), engine will show M (628) and entity A is verified. Otherwise, error message will be shown if sender (in this case entity A) is not verified (627).

In cases where message is signed (501, 601) and verified (521, 621), entity B can authenticate the sender as entity A using public key, $K_{pbA}$ attached to the message or from its own contact book. Authentication of entity A is carried out by having the application comparing the public key appended to the plain text against entity A's public key, $K_{pbA}$ held in entity B's mobile device. When the identity of entity B is authenticated and integrity of the sent message is verified by the fresh hash result, the sent message cannot be repudiated, thus produces non-repudiation of the message.

While the invention has been described in connection with a preferred embodiment, it is not intended to limit the scope of the invention to the particular form set forth, but on the contrary, it is intended to cover such alternatives, modifications, and equivalents as may be included within the spirit and scope of the invention as defined by the appended claims.

## Claims

1. A method for communicating in a secure manner with a mobile device comprising the steps of:

generating a first entity's public key and a first entity's private key on said mobile device belonging to the first entity;

generating a second entity's public key and a second entity's private key on a telecommunication device of the second entity;

exchanging the first entity's public key with the second entity' public key;

generating an encryption key on said adapted mobile device by using the second entity's public key and the first entity's private key according to an asymmetric cryptography algorithm;

encrypting a message to be sent by the first entity to the second entity with the encryption key by using a symmetric cryptography algorithm;

generating a decryption key on the telecommunication device by using the first entity's public key and the second entity's private key according to the asymmetric cryptography algorithm; and

decrypting the encrypted message sent by the first entity to the second entity with the decryption key by using the symmetric cryptography algorithm.

2. The method as claimed in claim 1 wherein the steps of generating respective said entity's public and private key, said encryption key and said decryption key is carried out by means of Elliptic Curve Cryptography.

3. The method as claimed in claim 1 wherein the steps of encrypting the message and decrypting the message are carried out by means of Advance Encryption Standard.

4. The method as claimed in claim 1 wherein the steps of encrypting a message and decrypting a message are carried out by means of a proprietary 32-bit base symmetric cryptography algorithm.

5. The method as claimed in claim 1 wherein the respective steps of encrypting the message and decrypting the message are carried out according to specified encryption level

information sent together with the encrypted message.

6. The method as claimed in claim 1 further comprising the step of encoding the encrypted message on the mobile device so that said encrypted message is not dropped along its transmission in the telecommunication network and the step of decoding the encoded encrypted message on the telecommunication device.

7. The method as claimed in claim 1 wherein the telecommunication device is another mobile device.

8. The method as claimed in claim 1 wherein the telecommunication device is a computer.

9. The method as claimed in claim 8 wherein the computer is a server.

10. The method as claimed in claim 1 wherein the step of exchanging the public keys is carried out by one of the entities directly requesting the other entity's public key from the other entity.

11. The method as claimed in claim 1 wherein the step of exchanging the public keys is carried out by one of said entities storing its said public key in a server and the other said entity requesting said stored public key from said server.

12. The method as claimed in claim 1 wherein the step of exchanging the public key further comprising steps of: registering respective said entity's information, said information includes respective said entity's telephone number, respective said entity's name and respective said entity's public key; storing the entities' public keys in a key repository at the server; receiving a request for one of said entity's public key from another said entity; and sending the requested public key to said entity that requested said public key.

13. The method as claimed in claims 11 and 12 further comprising the step of authenticating said entity that requested said public key.

14. The method as claimed in claim 13 wherein the step of authenticating comprising the

steps of sending said requesting entity's public key to the server and verifying said sent public key against said requesting entity's public key that was stored earlier in the server.

15. The method as claimed in claim 1 wherein said second entity is a server and said server exchanges its said public key to other said entities by broadcasting its said public keys to all the entity.

16. The method as claimed in claim 1 wherein said second entity is a server and said server dispatches its said public key to a new entity added to the database of said server.

17. The method as claimed in claim 1 wherein said second entity is a server and said server uploads it public key to the repository.

18. A method for communicating in a trusted manner with a mobile device comprising the steps of: signing a message sent from the mobile device belonging to a first entity; and verifying the signed message received by a second entity's telecommunication device.

19. The method for communicating in a trusted manner with a mobile device as claimed in claim 18 wherein the step of signing the message comprising the steps of generating the first entity's public key and a first entity's private key on the mobile device belonging to the first entity, generating a hash result of the message on said adapted mobile device, generating a digital signature of said message by using the first entity's private key and said hash result, and attaching said digital signature and the first entity's public key to said message.

20. The method for communicating in a trusted manner with a mobile device as claimed in claim 18 wherein the step verifying the signed message comprising the steps of generating a fresh hash result of the signed message on said telecommunication device, generating an output value R of said message by using the first entity's public key, the digital signature and the fresh hash result; and comparing the output value R with an original value r in the digital signature, wherein said message is verified when the output value R is the same as the original value r.

21. The method for communicating in trusted manner with a mobile device as claimed in claim 20 further comprising the step of authenticating the signed message wherein said step of authenticating comprising the steps of providing a copy of the first entity's public key to the second entity, the copy of first entity's public key is provided at a moment different from the moment the signed message is sent, and comparing the first entity's public key in the signed message with the copy of first entity's public key, wherein said message is authenticated when the public key in the message is the same as the copy of the public key provided to the second entity.

22. The method for communicating in a trusted manner with a mobile device as claimed in claims 19 and 21 wherein said steps of generating said public key and said private key and generating said digital signature and said output value R respectively is according to Elliptic Curve Digital Signature Algorithm.

23. The method for communicating in a trusted manner with a mobile device as claimed in claims 19 and 21 wherein said steps of generating respectively said hash result and said fresh hash result is according to Secure Hash Algorithm.

24. The method for communicating in trusted manner with a mobile device as claimed in claim 18 further comprising the steps of encrypting and decrypting of said signed message according to claims 1 to 17.

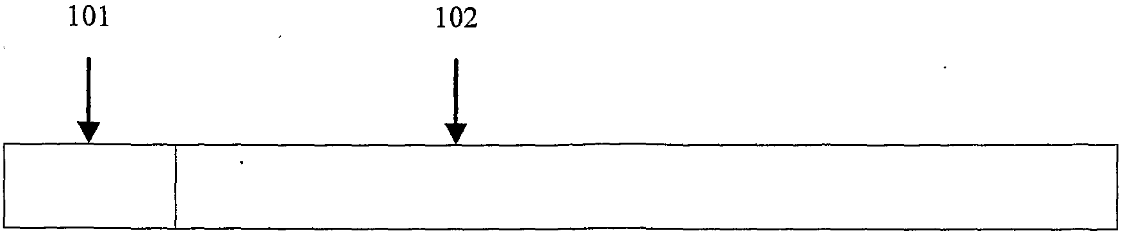


Figure 1a

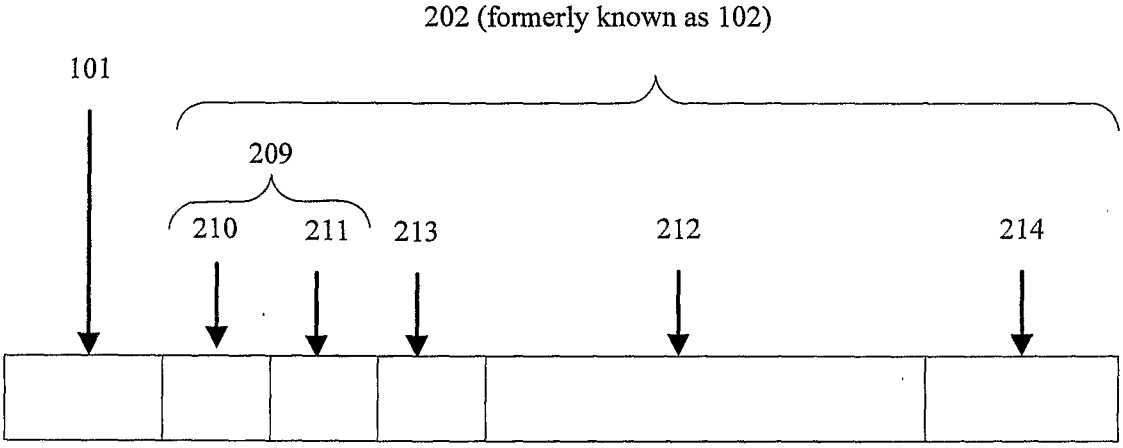


Figure 1b

| **Options for encryption/signing with different types of text representation** | **Maximum size of an SMS in characters** | **Maximum size for SMM type message with AES algorithm** | **Maximum size for SMM type message with novel proprietary cryptographic algorithm** |
|---|---|---|---|
| Only Encryption with 7 bits representation for both xKey and password | 160 | 112 | 112 |
| Signing and encryption with 7 bits representation | 160 | 48 | 60 |
| Only Signing with 7 bits representation | 160 | 80 | 84 |
| Only encryption with 8 bits representation | 140 | 96 | 96 |
| Signing and encryption with 8 bits representation | 140 | 48 | 52 |
| Only Signing with 8 bits representation | 140 | 64 | 72 |
| Only encryption with 16 bits unicode representation | 70 | 32 | 44 |
| Signing and encryption with 16 bits unicode representation | 70 | 16 | 24 |
| Only signing with 16 bits unicode representation | 70 | 32 | 32 |

Figure 2

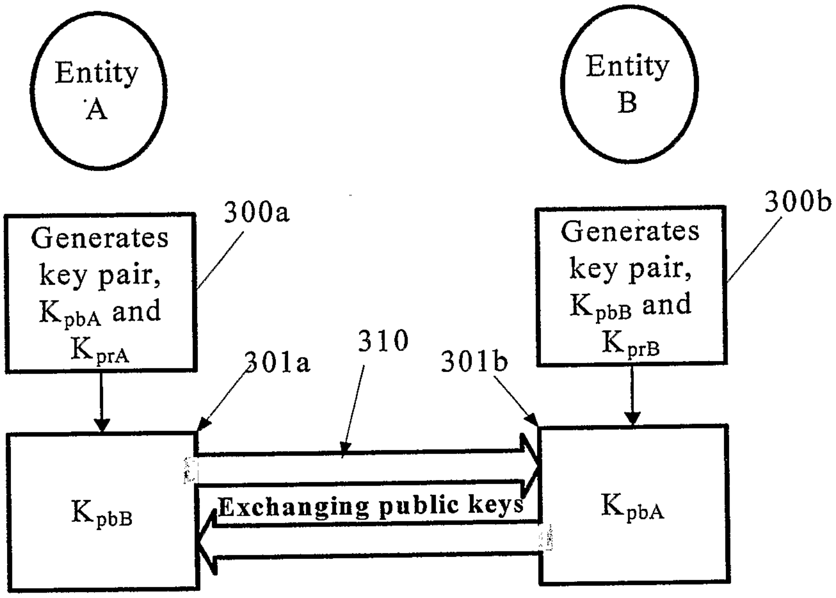

Entity A
Entity B
Generates key pair, $K_{pbA}$ and $K_{prA}$
300a
Generates key pair, $K_{pbB}$ and $K_{prB}$
300b
310
301a
301b
$K_{pbB}$
Exchanging public keys
$K_{pbA}$

Figure 3a

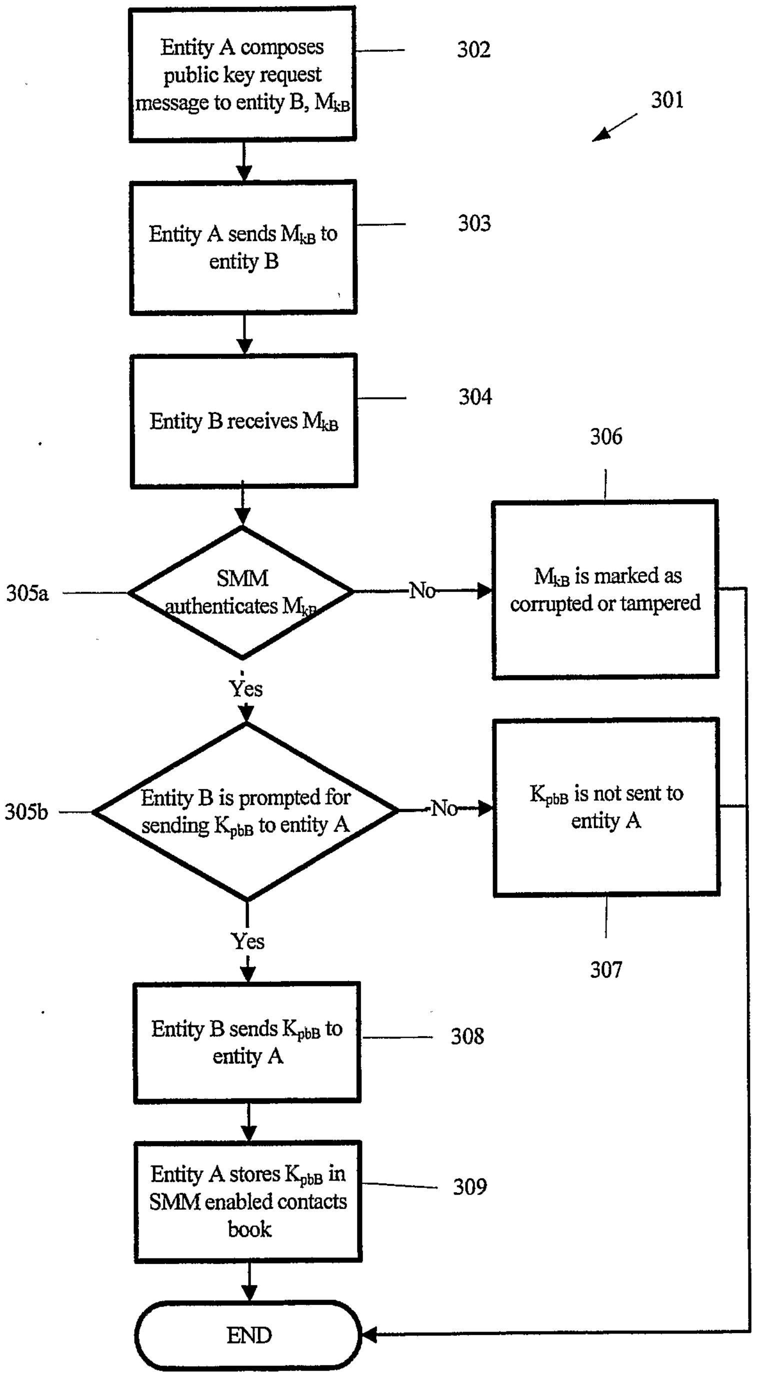

Entity A composes public key request message to entity B, $M_{kB}$
302
301
Entity A sends $M_{kB}$ to entity B
303
Entity B receives $M_{kB}$
304
306
305a
SMM authenticates $M_{kB}$
No
$M_{kB}$ is marked as corrupted or tampered
Yes
305b
Entity B is prompted for sending $K_{pbB}$ to entity A
No
$K_{pbB}$ is not sent to entity A
Yes
307
Entity B sends $K_{pbB}$ to entity A
308
Entity A stores $K_{pbB}$ in SMM enabled contacts book
309
END

Figure 3b

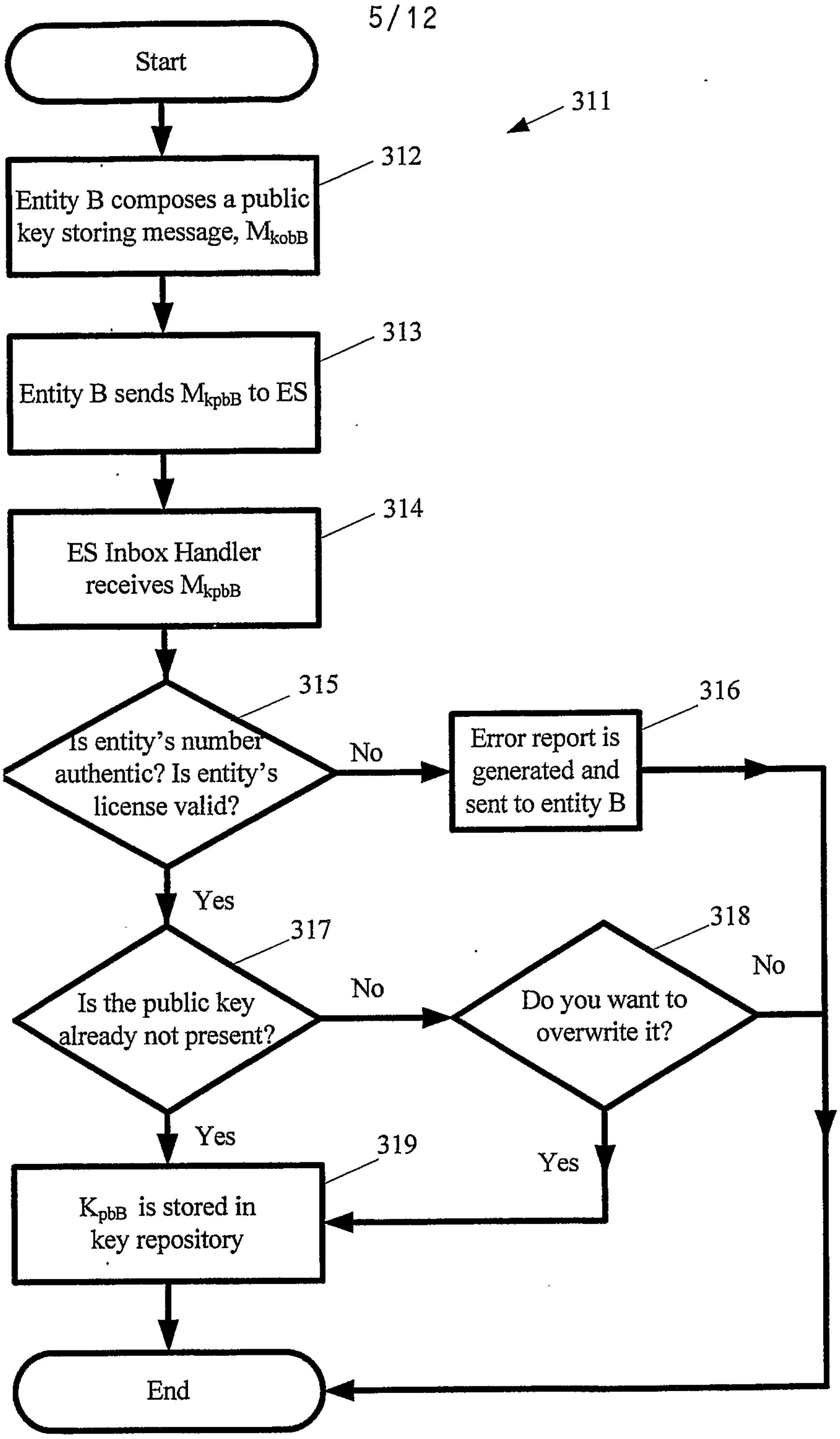

Start
311
312
Entity B composes a public key storing message, $M_{kobB}$
313
Entity B sends $M_{kpbB}$ to ES
314
ES Inbox Handler receives $M_{kpbB}$
315
Is entity's number authentic? Is entity's license valid?
No
316
Error report is generated and sent to entity B
Yes
317
Is the public key already not present?
No
318
Do you want to overwrite it?
No
Yes
319
Yes
$K_{pbB}$ is stored in key repository
End

Figure 3c

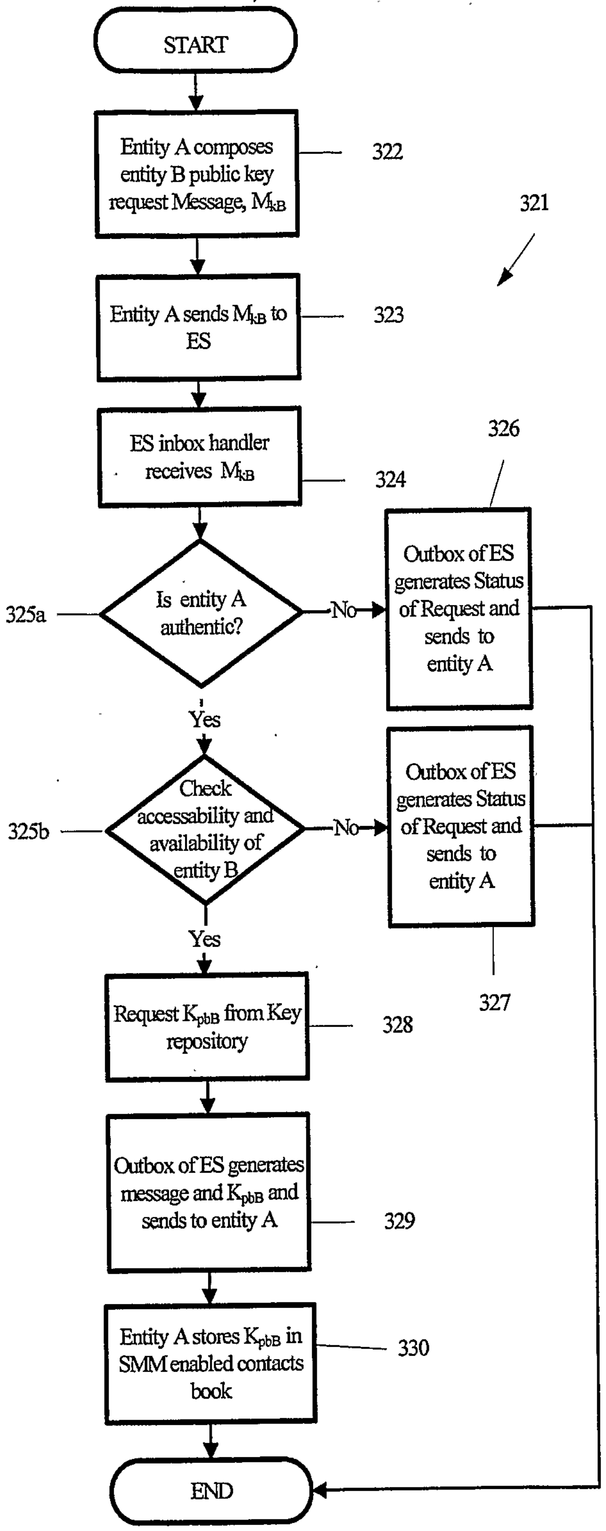

START
Entity A composes entity B public key request Message, $M_{kB}$
322
321
Entity A sends $M_{kB}$ to ES
323
ES inbox handler receives $M_{kB}$
324
326
Is entity A authentic?
325a
No
Outbox of ES generates Status of Request and sends to entity A
Yes
Check accessability and availability of entity B
325b
No
Outbox of ES generates Status of Request and sends to entity A
327
Yes
Request $K_{pbB}$ from Key repository
328
Outbox of ES generates message and $K_{pbB}$ and sends to entity A
329
Entity A stores $K_{pbB}$ in SMM enabled contacts book
330
END

Figure 3d

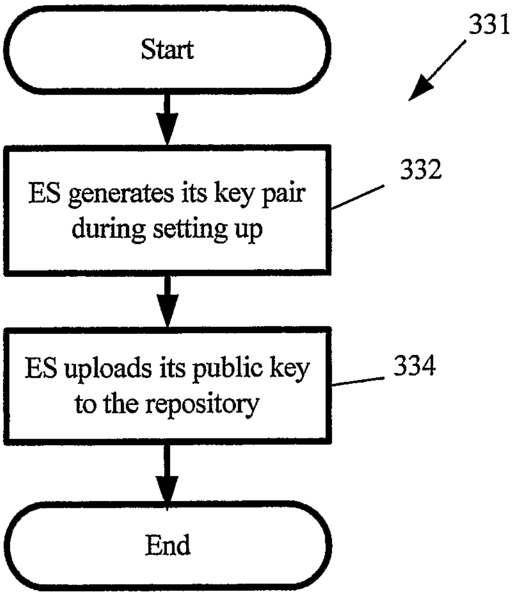

331
Start
ES generates its key pair during setting up
332
ES uploads its public key to the repository
334
End

Figure 3e

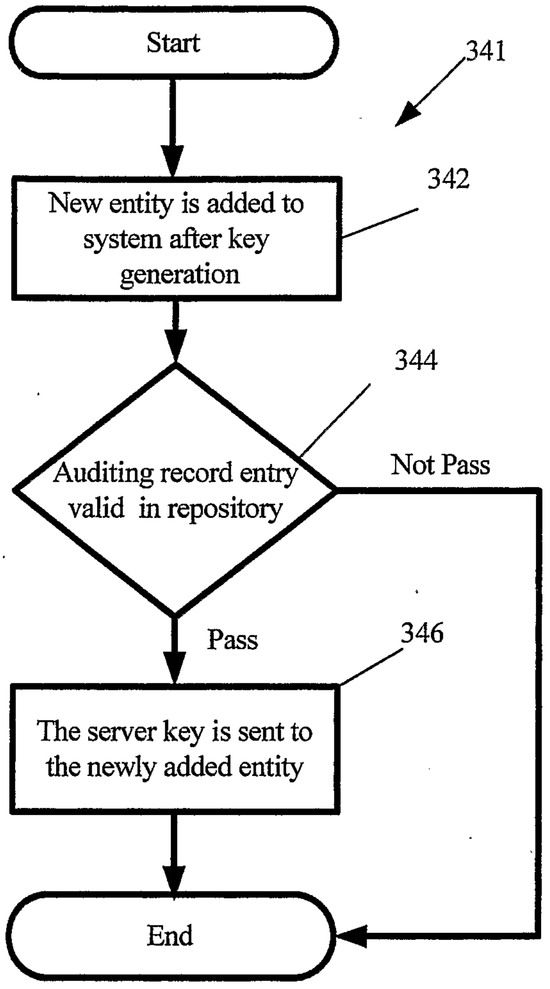

Start
341
New entity is added to system after key generation
342
344
Auditing record entry valid in repository
Not Pass
Pass
346
The server key is sent to the newly added entity
End

Figure 3f

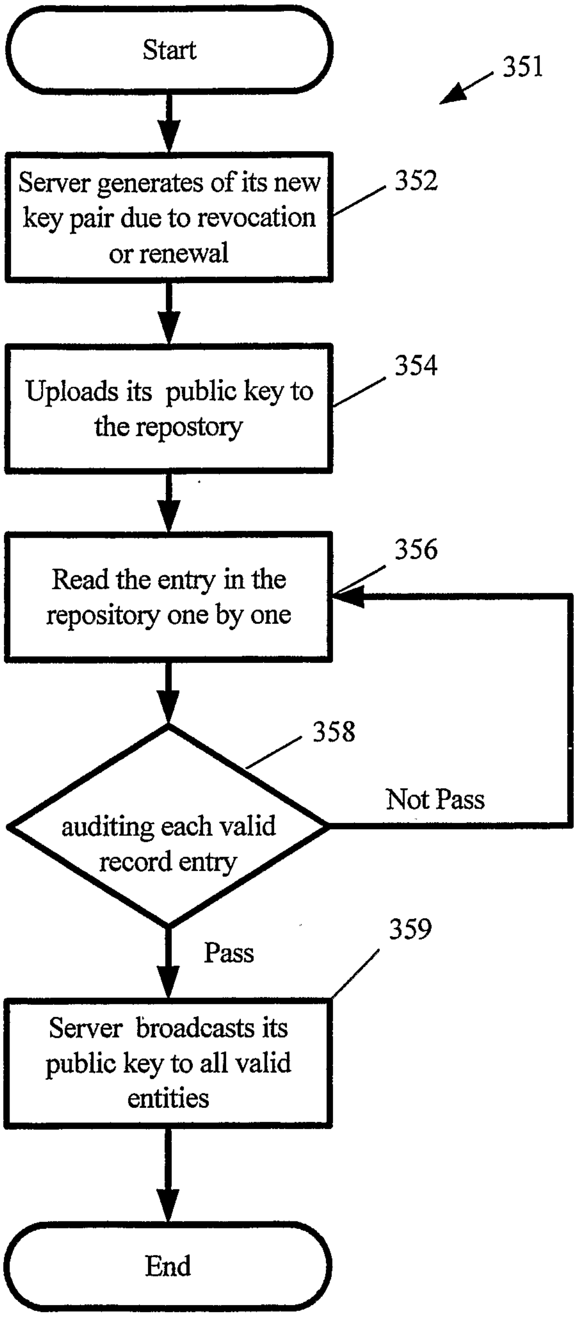

Start
351
Server generates of its new key pair due to revocation or renewal
352
Uploads its public key to the repostory
354
Read the entry in the repository one by one
356
auditing each valid record entry
358
Not Pass
Pass
359
Server broadcasts its public key to all valid entities
End

Figure 3g

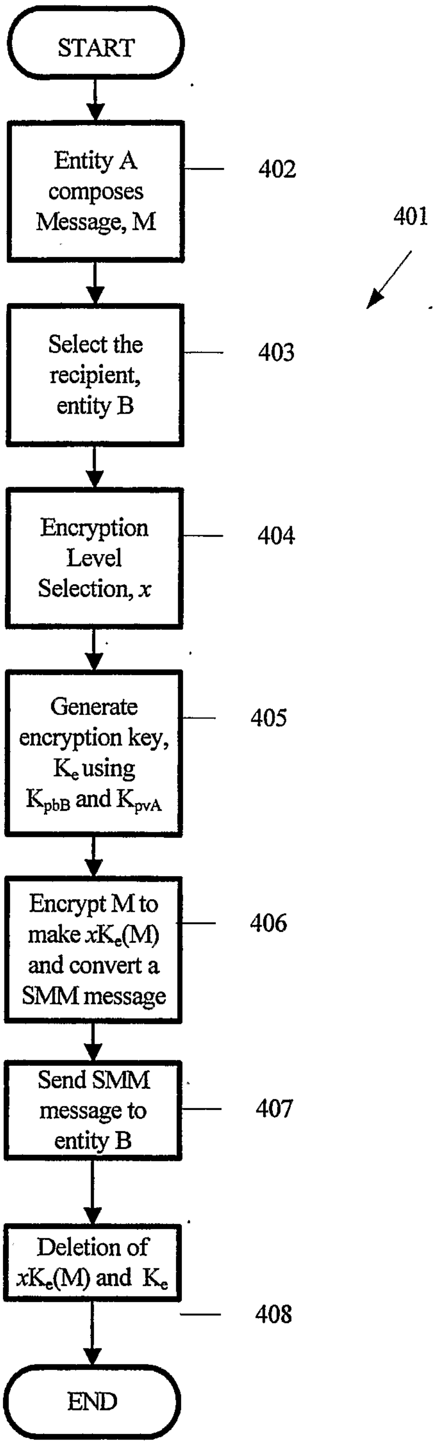

START
Entity A composes Message, M
402
401
Select the recipient, entity B
403
Encryption Level Selection, $x$
404
Generate encryption key, $K_e$ using $K_{pbB}$ and $K_{pvA}$
405
Encrypt M to make $xK_e(M)$ and convert a SMM message
406
Send SMM message to entity B
407
Deletion of $xK_e(M)$ and $K_e$
408
END

Figure 4a

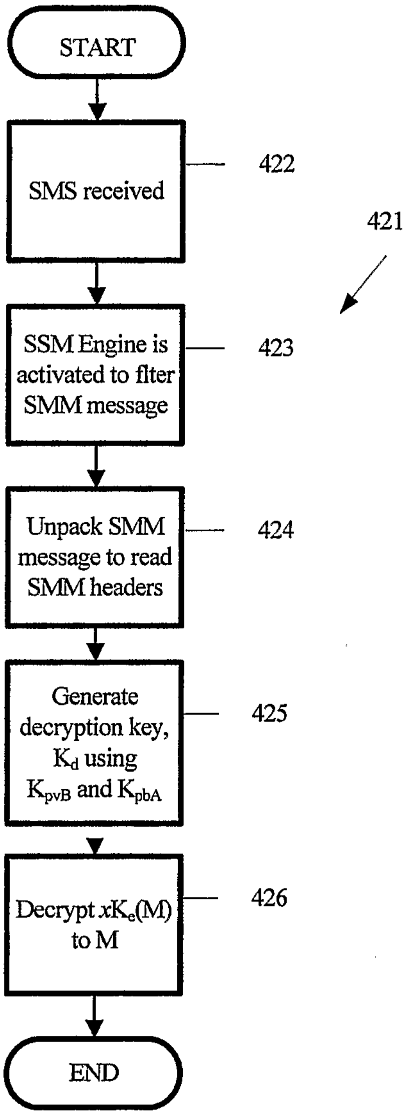

START
SMS received
422
421
SSM Engine is activated to flter SMM message
423
Unpack SMM message to read SMM headers
424
Generate decryption key, $K_d$ using $K_{pvB}$ and $K_{pbA}$
425
Decrypt $xK_e(M)$ to M
426
END

Figure 4b

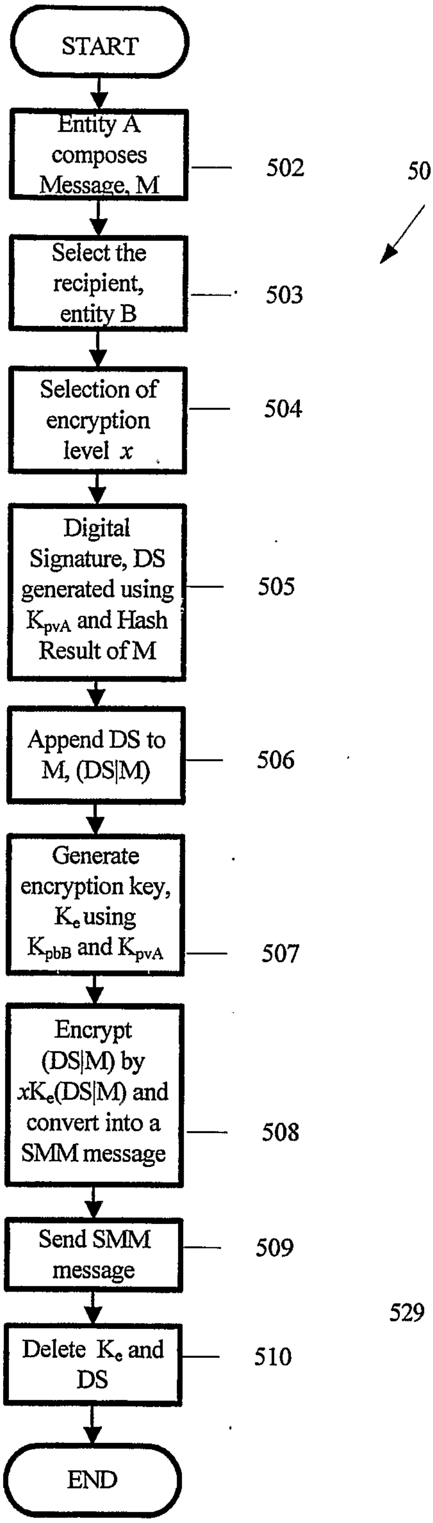

START
Entity A composes Message, M
502
501
Select the recipient, entity B
503
Selection of encryption level $x$
504
Digital Signature, DS generated using $K_{pvA}$ and Hash Result of M
505
Append DS to M, (DS|M)
506
Generate encryption key, $K_e$ using $K_{pbB}$ and $K_{pvA}$
507
Encrypt (DS|M) by $xK_e$(DS|M) and convert into a SMM message
508
Send SMM message
509
Delete $K_e$ and DS
510
END

Figure 5a

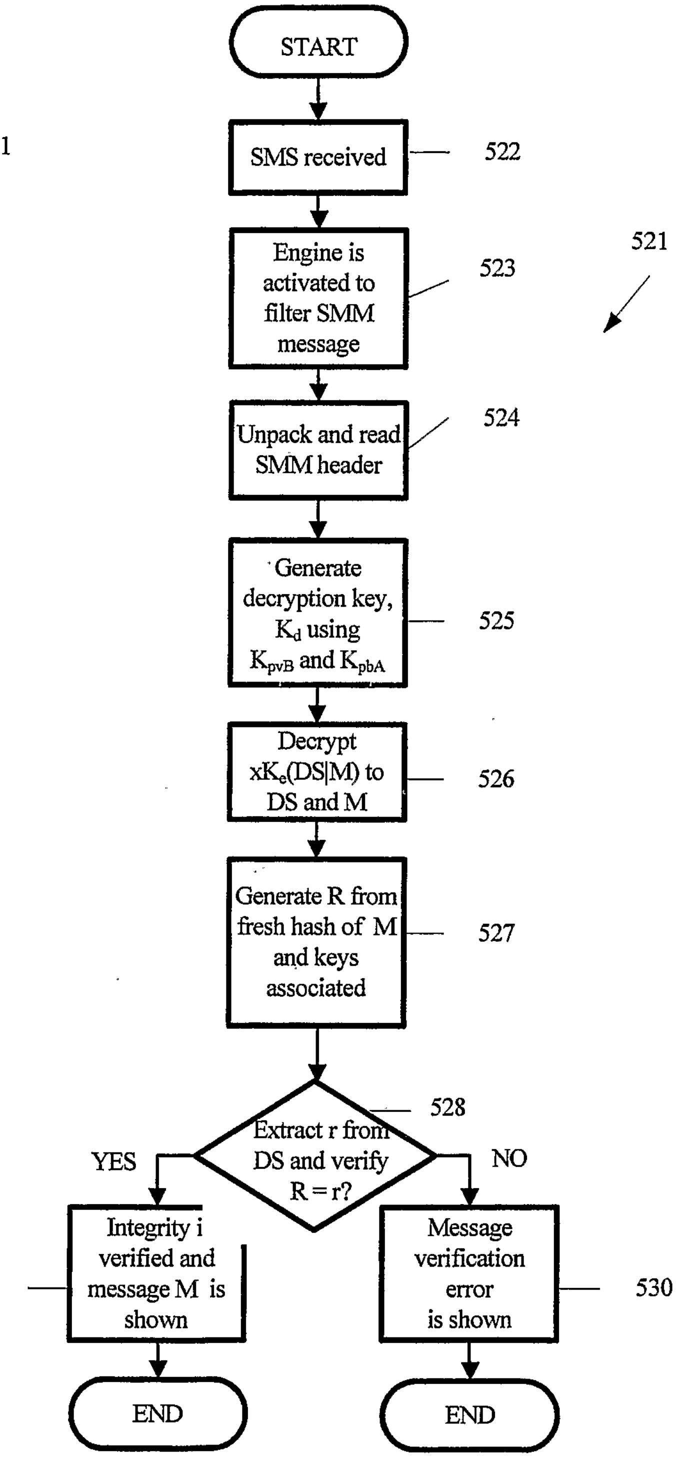

START
SMS received
522
521
Engine is activated to filter SMM message
523
Unpack and read SMM header
524
Generate decryption key, $K_d$ using $K_{pvB}$ and $K_{pbA}$
525
Decrypt $xK_e$(DS|M) to DS and M
526
Generate R from fresh hash of M and keys associated
527
Extract r from DS and verify R = r?
528
YES
NO
Integrity i verified and message M is shown
529
Message verification error is shown
530
END
END

Figure 5b

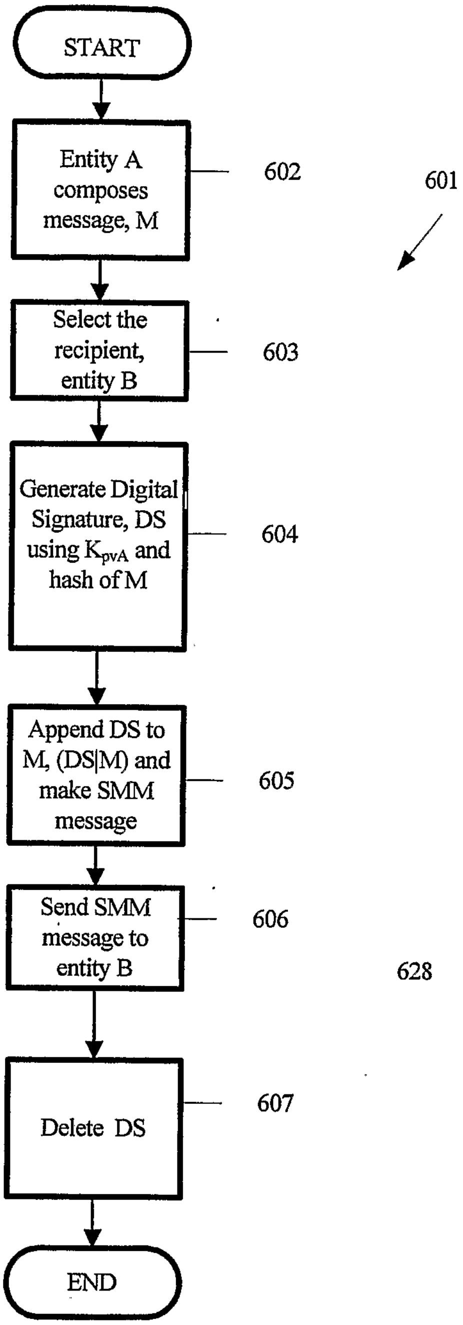

START
Entity A composes message, M
602
Select the recipient, entity B
603
Generate Digital Signature, DS using $K_{pvA}$ and hash of M
604
Append DS to M, (DS|M) and make SMM message
605
Send SMM message to entity B
606
Delete DS
607
END
601

Figure 6a

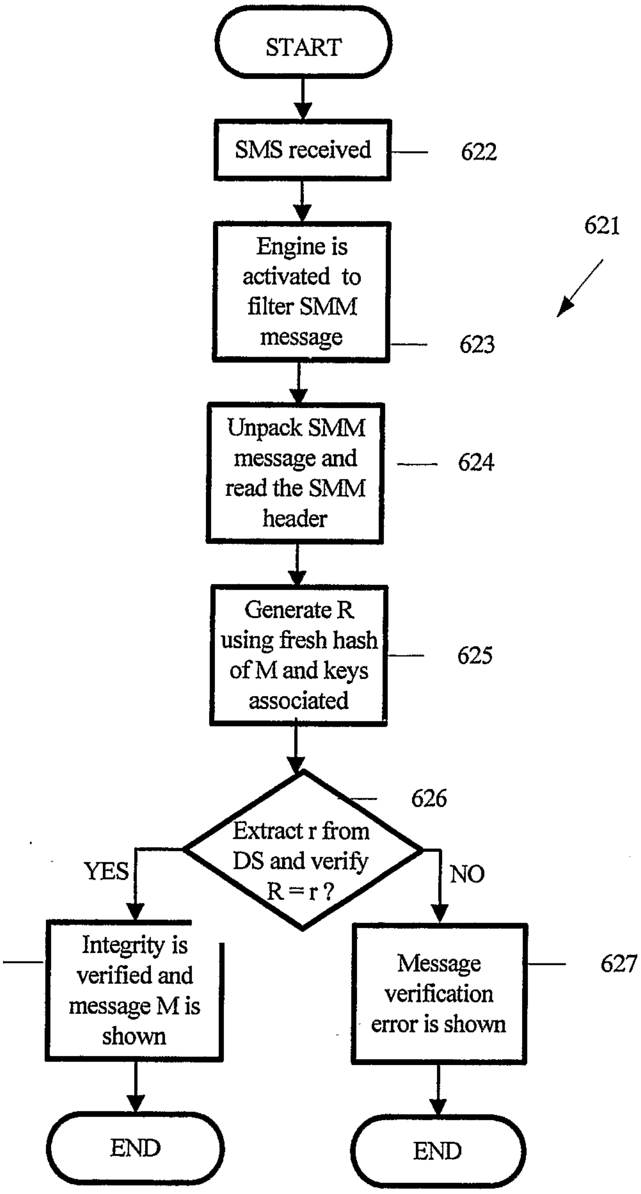

START
SMS received
622
Engine is activated to filter SMM message
623
621
Unpack SMM message and read the SMM header
624
Generate R using fresh hash of M and keys associated
625
Extract r from DS and verify R = r ?
626
YES
NO
628
Integrity is verified and message M is shown
Message verification error is shown
627
END
END

Figure 6b

|  |  |
|---|---|
| **INTERNATIONAL SEARCH REPORT** | International application No.<br>**PCT**/SG2006/000229 |

A. CLASSIFICATION OF SUBJECT MATTER

Int. Cl.

***H04L 9/30*** (2006.01)

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)

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)
dwpi, uspats, usapps, wopatents, epapat, epbpat, japio, uspto, internet: Keywords-Encrypt, encipher, encode, key, public, private, asymmetric, algorithm, function, exchane, mobile, wireless, elliptic and similar terms

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| Y | MASON, "IPSEC OVERVIEW PART THREE: CRYPTOGRAPHIC TECHNOLOGIES" 22 February 2002, Retrieved from the internet from: *URL:www.ciscopress.com/articles/printerfriendly.asp?p=25473&r1=1&r1=1* on 13 September 2006 | 1 - 17 |
| Y | US 20050010801 A1 (SPIES et al) 13 January 2005, Abstract, paras 29, 31, 44, and 45 | 1-17 |
| Y | US 20020123967 A1 (WANG) 5 September 2002 Abstract, paras 38, 115 | 1-17 |
| Y | US 20040203581 A1 (SHARON et al) 14 October 2004 Abstract, paras 33, and 65 | 3 |

☒ 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<br>13 September 2006 | Date of mailing of the international search report<br>6 NOV 2006 |
| Name and mailing address of the ISA/AU<br>AUSTRALIAN PATENT OFFICE<br>PO BOX 200, WODEN ACT 2606, AUSTRALIA<br>E-mail address: pct@ipaustralia.gov.au<br>Facsimile No. (02) 6285 3929 | Authorized officer<br>**ROBERT BARTRAM**<br>Telephone No : (02) 6283 2215 |

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | WONG et al, "MUTUAL AUTHENTICATION AND KEY EXCHANGE FOR LOW POWER WIRELESS COMMUNICATIONS", Military communications conference, 2001, MILCOM 2001Network-Centric operations: Creating the information force. IEEE Vol 1 pg 39-43 | 1-17 |
| A | SCHNEIER B, "APPLIED CRYPTOGRAPHY, SECOND EDITION" John Wiley and Sons Inc,© 1996, pgs 4, 5, 30, 31, 34,-44, 151, 152, 480, 481, 486-489, 513, 514 | 1-17 |
|  | Note: for Y documents Mason is combined with anyone of the following three documents, US 20050010801, US 20020123967, or US 20040203581. |  |

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.
**PCT/SG2006/000229**

This Annex lists the known "A" publication level patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

| Patent Document Cited in Search Report | | Patent Family Member | | | | | |
|---|---|---|---|---|---|---|---|
| US | 2005010801 | EP | 1636933 | US | 7017181 | WO | 2005001629 |
| US | 2002123967 | AU | 20597/01 | AU | 40043/00 | AU | 53831/98 |
| | | AU | 2002247213 | AU | 2002357047 | AU | 2002367640 |
| | | CA | 2365644 | CA | 2403332 | CN | 1344396 |
| | | CN | 1452739 | CN | 1623173 | EP | 1159700 |
| | | EP | 1272933 | US | 5917913 | US | 6175922 |
| | | US | 6282656 | US | 6594759 | US | 6850916 |
| | | US | 7089214 | US | 7107246 | US | 2002023215 |
| | | US | 2003004827 | WO | 0052866 | WO | 0169388 |
| | | WO | 9825371 | WO | 02069291 | WO | 03065318 |
| | | WO | 03081377 | | | | |
| US | 2004203581 | AU | 2003269443 | WO | 2004032451 | | |

Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001.

END OF ANNEX