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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
31 May 2012 (31.05.2012)

WIPO | PCT

(10) International Publication Number
WO 2012/070922 A1

(51) International Patent Classification:
G06F 21/00 (2006.01) *H04L 9/08* (2006.01)

(21) International Application Number:
PCT/MY2011/000069

(22) International Filing Date:
3 June 2011 (03.06.2011)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
PI2010005503 23 November 2010 (23.11.2010) MY

(71) Applicant *(for all designated States except US)*: **MIMOS BERHAD** [MY/MY]; Technology Park Malaysia, 57000 Kuala Lumpur (MY).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **SEA, Chong Seak** [MY/MY]; SDCE, Mimos Berhad, Technology Park Malaysia, 57000 Kuala Lumpur (MY). **NG, Kang Siong** [MY/MY]; SDCE, Mimos Berhad, Technology Park Malaysia, 57000 Kuala Lumpur (MY).

(74) Agent: **HONG, Lok Choon**; Pintas IP Group SDN BHD, Suite 6.03, 6th Floor, Wisma Mirama, Jalan Wisma Putra, 50460 Kuala Lumpur (MY).

(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, RO, RS, RU, 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, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ,

[Continued on next page]

(54) Title: A METHOD OF CONTROLLING LICENSE KEY GENERATION

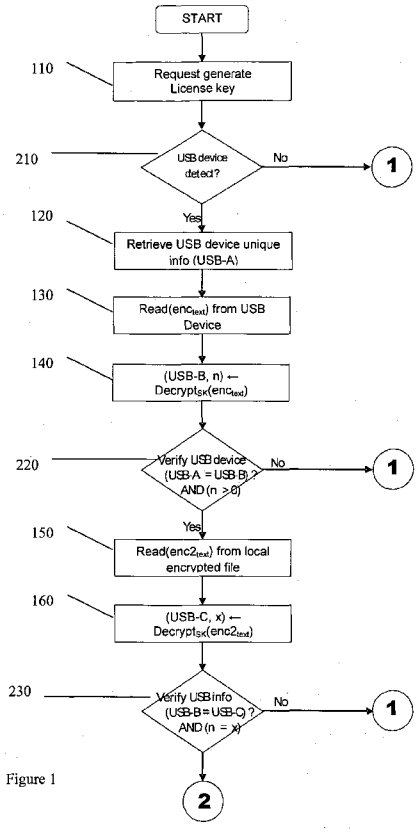

Figure 1

(57) Abstract: A method of generating software license connecting security token and making request to client system via communication module that security token contains first verification information pack, encrypted using secret key, including primary digital identity of software vendor generated based on hardware information of security token. First digital counter; extracting hardware information of security token to generate secondary digital identity; decrypting first verification information pack using secret key to gain access of primary digital identity and digital counter; verifying primary digital identity is similar to secondary digital identity; decrypting second verification information pack, containing tertiary digital identity and second digital counter; generating copy of license key upon verifying the primary and tertiary digital identities are same; generating third digital counter; encrypting third digital counter together with primary or tertiary digital identity using secret key to produce third verification information pack; replacing first and second verification information pack with third verification information pack.

WO 2012/070922 A1

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))*

A METHOD OF CONTROLLING LICENSE KEY GENERATION

FIELD OF INVENTION

The present invention relates to a method or mechanism employed to control authorized use of a software, particularly limiting copy of license keys can be generated by a known vendor or reseller.

BACKGROUND OF THE INVENTION

Piracy is a significant problem leading to billion of income lose in the software industry annually. Different approaches and measurements have been implemented in the industry to prevent unauthorized installation, use or copy of a software particular at the end user side. For example, United State patent application no 5864620 discloses a system and method for controlling distribution of software to an user in a multitiered distribution chain that a license clearing house is established in the system to be responsible for controlling usage rights of a software. The license clearing house sends out reply containing authentication certificate and a master key once validated request sent by the user. Another method of software verification is described in United States patent application no 6898286 which validates the product license using a verification code generated from the hardware of the computer installed with the software.

Inoue discloses another system to manage number of software installed in a computer network in another United States patent application no. 2003/0061136. The system particularly involves a license management apparatus in communication with a computer which send a request the license management apparatus to acquire approval to use the software, while the license management provide the approval as long the software concurrently used is less than a maximum licensed. The license management apparatus in the disclosed system also closely monitor the number of computer using the software by actively requesting update from the connected computer. Further license management system is provided in United State patent with publication no. 2010/0031372 to control number of subset computer allowed to used the software. A

peripheral licensing is used in the system dedicated to store licensing information while storing of such information into the device is prevented once the number of subset computer using the software exceeds the authorized number.

However, the abovementioned system and management system are particular aiming to avoid piracy at the user end while, in fact, piracy can be performed as well at the reseller or distributor end especially the reseller end is offered the right to generate authentic licensing key to activate use of a software. This type of multitier distribution system may jeopardize interest of the software author as the software author may be exposed to the risk of not being informed about extra generated license key and no receiving any royalty thereof from the reseller. Thus, it is important to have an effective mechanism to render the software author the right to control license activation key generated by the reseller in order to protect interest of the software author.

SUMMARY OF THE INVENTION

The present invention aims to provide a mechanism to monitor and control the amount of license key generated by a known software reseller or distributor from a client system of the software distribution network.

Another object of the present invention is to provide a license and royalty management system to protect interest of the software author. The reseller and/or distributor may have to pay the royalty owing to the software author before the reseller and/or distributor can acquire license key from the client system upon finished the previous allocated amount.

Further object of the present invention is to offer a mechanism to monitor and control the amount of license key generated by a known software reseller using a down counter to limit the number of license can be generated thereof through the client system. Identical copy of down counter is stored in a database of the client system for verification purpose and it is employed as an approach to counter tampering of the recorded copy in the security token.

At least one of the preceding objects is met, in whole or in part, by the present invention, in which one of the embodiment of the present invention involves a method of generating a software license key for a software vendor in a client system comprising the steps of connecting a security token and making request of license key generation to the client system through a communication module that the security token contains a first verification information pack, encrypted using a secret key, including a primary digital identity of the software vendor generated based on hardware information of the security token and a first digital counter having an integer N; extracting hardware information of the security token to generate a secondary digital identity; decrypting the first verification information pack using the secret key to gain access of the primary digital identity and the digital counter; verifying the primary digital identity is similar to the secondary digital identity and N is not equal to zero; decrypting a second verification information pack, which containing a tertiary digital identity and a second digital counter having an integer X; generating Z copy of license key that Z is smaller than N or X upon verifying that the primary and tertiary digital identities are the same while N equals to X; generating a third digital counter having an integer of (N-Z) or (X-Z); encrypting the third digital counter together with the primary or tertiary digital identity using the secret key to produce a third verification information pack; and replacing the first and second verification information pack with the third verification information pack.

In another aspect of the disclosed method, the first verification information pack is provided into the security token via the steps comprising of connecting the security token to the client system through a communication module; retrieving the hardware information to generate the primary digital identity; encrypting the primary digital identity with the integer N provided by the client system, using the secret key, to produce the first verification information pack; and storing the first verification information pack on the security token.

Still another aspect of the method includes additional step of storing the first verification information pack into a database of the client system as the second verification information pack.

Preferably, the security token of the present invention is a smart card, USB (Universal Serial Bus) storage device or smart-card-based USB device, while the communication module is bluetooth, RFID (Radio Frequency Identification), or USB coupling.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows one embodiment of the disclosed method to verify the license key generation by the reseller through a client system;

Figure 2 shows one embodiment of the disclosed method in generating the license key upon verifying the authenticity of the connected security token;

Figure 3 show a sub-sequence in the disclosed method which inform rejection of license key generation upon failing the verification process shown in figure 1; and

Figure 4 shows a sub-sequence of the disclosed method in setting up or refreshing the security token together with embedding the counter into the security token.

DETAILED DESCRIPTION OF THE INVENITON

One skilled in the art will readily appreciate that the present invention is well adapted to carry out the objects and obtain the ends and advantages mentioned, as well as those inherent therein. The embodiment describes herein is not intended as limitations on the scope of the invention.

The present invention includes a method, as shown in both figure 1 and 2, of generating a software license key for a software vendor in a client system comprising the steps of connecting a security token and making request of license key generation (110) to the client system through a communication module that the security token contains a first verification information pack, encrypted using a secret key, including a primary digital identity of the software vendor generated based on hardware information of the security token and a first digital counter having an integer N;

extracting hardware information (120, 130) of the security token to generate a secondary digital identity; decrypting the first verification information pack (140) using the secret key to gain access of the primary digital identity and the digital counter; verifying (220) the primary digital identity is similar to the secondary digital identity and N is not equal to zero; decrypting (150, 160) a second verification information pack, which containing a tertiary digital identity and a second digital counter having an integer X; generating Z copy of license key (310) that Z is smaller than N or X upon verifying (230) that the primary and tertiary digital identities are the same while N equals to X; generating (330) a third digital counter having an integer of (N-Z) or (X-Z); encrypting the third digital counter together with the primary or tertiary digital identity using the secret key to produce a third verification information pack; and replacing (340, 350) the first and second verification information pack with the third verification information pack.

It is important to be noted the security token is preferably a device equipped with storage capacity in order to store the encrypted first verification information pack for reading and identity validation to prompt the client system to generate the license keys thereof. The first verification information pack contains encrypted information, the primary digital identity, which is device and vendor-specific. More specifically, the primary digital identity is generated mainly from serial number of the security token with or without in combination of other information which is device specific. In turn, the serial number of the security token is used as reference to identify the vendor assigned to this particular security token. The security token of the present can adapt many known formats available in the field. For example, it can be a smart card, a USB storage device or smart-card-based USB device as long sufficient storage space are available for reading, writing and storing the first verification information pack. Further, the security token of the present invention shall possess as well a communication module to make the stored information accessible for reading, writing and processing by the client system. The communication module can be either contact-based or contactless-based. Bluetooth, RFID, or GSM frequency may be used in the contactless-based communication module to exchange information, while USB coupling or other chips reading approaches are applicable in the present invention to facilitate communication establishment. Most preferably, USB-based module, but not

limited to, is employed in the present invention for identification, verification and communication purposes in the disclosed method that it is preferably illustrated in the embodiment shown in figure 1, 2 and 4.

Generally, the present invention can be divided into three major phases along the license key generation process. It firstly involves a series of process for verification of the connected security token, as shown in figure 1, followed by either an license generating phase as in figure 2 or rejection phase as in figure 3. Failing in any one of the verification steps leads to instant rejection phase and termination of the process loop, whereas license key is generated in the license generating phase with immediate update of information in the stored counter at both security token and client server upon success the verification process.

As in setting forth, the request to generate one or more license key from the client system is firstly made by establishing connection (110) with the client system using the security token. The connection herein can refer to communication with physical contact such as USB or contactless through bluetooth module. The client system conducts a preliminary verification (210) if the connecting security token was a recognizable format or not. Once the client system fails to recognize the format of the device wishing to make the connection, the failure status is reported (410) to the vendor through a message displayed on an user interface followed by termination and exit (420) of the process loop. Passing the preliminary verification subjects the connected security token to subsequent process which the client system retrieve unique information of the connected security token and use the unique information to generate a secondary digital identity using a secret key. The unique information is preferably serial number of the security token assigned to it by the manufacturer that partial of or complete serial number is used in generating (120) the secondary digital identity. The digital identity can directly derive from the serial number or being created through a specific algorithm using information derived from the serial number. Meanwhile, the client system read (130) the information available in the security token and begins the decryption (140) to acquire the primary digital identity and the first digital counter having an integer N upon detecting the encrypted information in the token. The client system performs another verification step at this

stage by comparing (220) the on spot-generated secondary digital identity to the decrypted primary digital identity. Verification is considered success if complete match is found or failure initiates failing message (410) with immediate exits (420) of the process loop. Additionally, the client system also trigger the rejection phase immediately once it finds the integer N is zero. Confirmation of the complete match and N is not Zero trigger the client system to read (150) and decrypt (160) the second verification information pack stored locally in the database of the client system. The encrypted second verification information pack contains a tertiary digital identity and a second digital counter having an integer X. Preferably, the encrypted file of the second verification information pack of the present invention is assigned with a name specific to the connected security token and/or the vendor. Preferably, name of the encrypted file of the second verification is created based on the unique information of the specific security token also. Such feature facilitates update or replacement of the old file with the latest one using the identical file name in both security token and database of the client system. With the decrypted data from the second verification information pack, the client system again matches the primary and tertiary digital identity for verifying the identity and authenticity of the security token. Also, the client system also match the integer X of the second digital counter to the integer N of the first digital counter that equal value of X and N corroborate to the authenticity of the user. Upon passing the verification step (230) at this stage, the process is directed to the acceptance phase, as in figure 2, where the client system generate the license key. Likewise, flunking the verification test at this step (230) initiates the failure message (410) followed by instant exit (420) of the process.

At the license generating phase, the client system generates (310) the license key requested by the vendor once the authentication process completed. Preferably, Z copies of license key are generated where the value of Z has to be smaller than N or X. User interface at the client system shall display an error message to the vendor if the value of Z is higher than N or X. Alternatively, the client system only allows one license key to be generated per session where Z = 1. The license key generated can either save into the security token as a new separate document or being sent to an user account belong to the vendor. The client generates (320) a third digital counter which has an integer of (N-Z) or (X-Z) once the license key generation is complete. The

client system then begins to encrypt (330) the third digital counter together with the primary digital identity or the tertiary digital identity to produce an encrypted third verification information pack. The newly produced third verification information pack possessing the updated digital counter is used to replace the first and second verification information pack of older version in both the security token and the database of client system. According to the preferred embodiment, the third verification information pack is stored to the security token (340) replacing the first verification information pack . Likewise, the third verification information pack is written (350) to the local database of the client system replacing the second verification information pack. A report may be delivered to user interface informing the vendor about the success status or the process may be returned (360) the verification process in order to generate further license key. The later scenario is employed particularly when the method has limited the number of key can be generated per session is one or Z = 1. Further, the client system may store another log report recording all the interaction in between the vendor and the client system as another counter measurement to track all the license key generation record.

In respect with another embodiment, the client system may have to format or set up the security token before the security token can be actually used for verification and recording purpose in the license key generation process. It is important that a new first verification information pack has to be created and stored into a new security token. Preferably, the first verification information pack is provided into the security token via the steps comprising of connecting (510) the security token to the client system through a communication module; retrieving the hardware information (530) to generate the primary digital identity; encrypting the primary digital identity with the integer N provided (520) by the client system, using the secret key, to produce the first verification information pack; and storing (550) the first verification information pack on the security token. More specifically, as in the foregoing description, the communication module can be of contact-based or contactless-based module depending on the format of the security token. The set up process started with establishing connection in between the client system and the security token. The client system, as illustrated in the figure 4, may have provided (520) a predetermined integer to the digital counter that the vendor has no privilege in interfering the value of the

integer. Optionally, a range of integer may be selectable to the vendor through the user interface available at the client system along the set up process (not shown in any drawing) that the optimal value of the integer may be varied from one vendor. For example, vendors whom have deposited money to the software author may be given much higher value to the contrary. Nevertheless, before any actual process can be performed to the security token, the client system has to first verify (240) the connected security token is in recognizable and/or authorized format. Device fails the verification lead to direct exits (570) of the process loop, preferably with an error message displayed on the user interface to inform the vendor of such failure. Upon verification, the client system retrieve (530) the unique information of the security token, for example serial code of the token, to generate a first digital identity. The first digital identity can be solely or partially derived from the retrieved unique information or even created from an algorithm using the retrieved information. Together the first digital counter with integer N and the first digital identity are encrypted (540) by the client system using a secret key, which can be one or more algorithm originated, to produce the first verification information pack. The client then stores (550) the created first verification information pack into the security token. Preferably, the client cater a storing or writing status report (560) to the vendor informing failure or success of the set up. Finally, the process loop exits (570) upon complete of the set up process.

In one embodiment, used security token is re-formatted by the client system when the digital counter equals to zero or the vendor is given a new/pre-formatted security token from the in-charge personnel of the client system in exchange of the used security token without the need of going through a top-up process.

More preferably the disclosed method also includes the step of storing the first verification information pack created in the setting up of the security token into a database of the client system that the stored information is applied as second verification information pack to be used for verification in the later stage of license key generation. This stored second verification information is tagged with a registered account of the vendor in the client system that the information is vendor-specific.

The present disclosure includes as contained in the appended claims, as well as that of the foregoing description. Although this invention has been described in its preferred form with a degree of particularity, it is understood that the present disclosure of the preferred form has been made only by way of example and that numerous changes in the details of construction and the combination and arrangements of parts may be resorted to without departing from the scope of the invention.

CLAIMS

1. A method of generating a software license key for a software vendor in a client system comprising the steps of

connecting a security token and making request of license key generation (110) to the client system through a communication module that the security token contains a first verification information pack, encrypted using a secret key, including a primary digital identity of the software vendor generated based on hardware information of the security token and a first digital counter having an integer N;

extracting hardware information (120, 130) of the security token to generate a secondary digital identity;

decrypting the first verification information pack (140) using the secret key to gain access of the primary digital identity and the digital counter;

verifying (220) the primary digital identity is similar to the secondary digital identity and N is not equal to zero;

decrypting (150, 160) a second verification information pack, which containing a tertiary digital identity and a second digital counter having an integer X;

generating Z copy of license key (310) that Z is smaller than N or X upon verifying (230) that the primary and tertiary digital identities are the same while N equals to X;

generating (330) a third digital counter having an integer of (N-Z) or (X-Z);

encrypting the third digital counter together with the primary or tertiary digital identity using the secret key to produce a third verification information pack; and

replacing (340, 350) the first and second verification information pack with the third verification information pack.

2. A method of claim 1, wherein the first verification information pack is provided into the security token via the steps comprising of

connecting (510) the security token to the client system through a communication module;

retrieving the hardware information (530) to generate the primary digital identity;

encrypting the primary digital identity with the integer N provided (520) by the client system, using the secret key, to produce the first verification information pack; and

storing (550) the first verification information pack on the security token.

3. A method of claim 2 further comprising step of storing the first verification information pack into a database of the client system as the second verification information pack.

4. A method of claim 1, wherein the security token is a smart card, USB storage device or smart-card-based USB device.

5. A method of claim 1 or 2, wherein the communication module is bluetooth, RFID, or USB coupling.

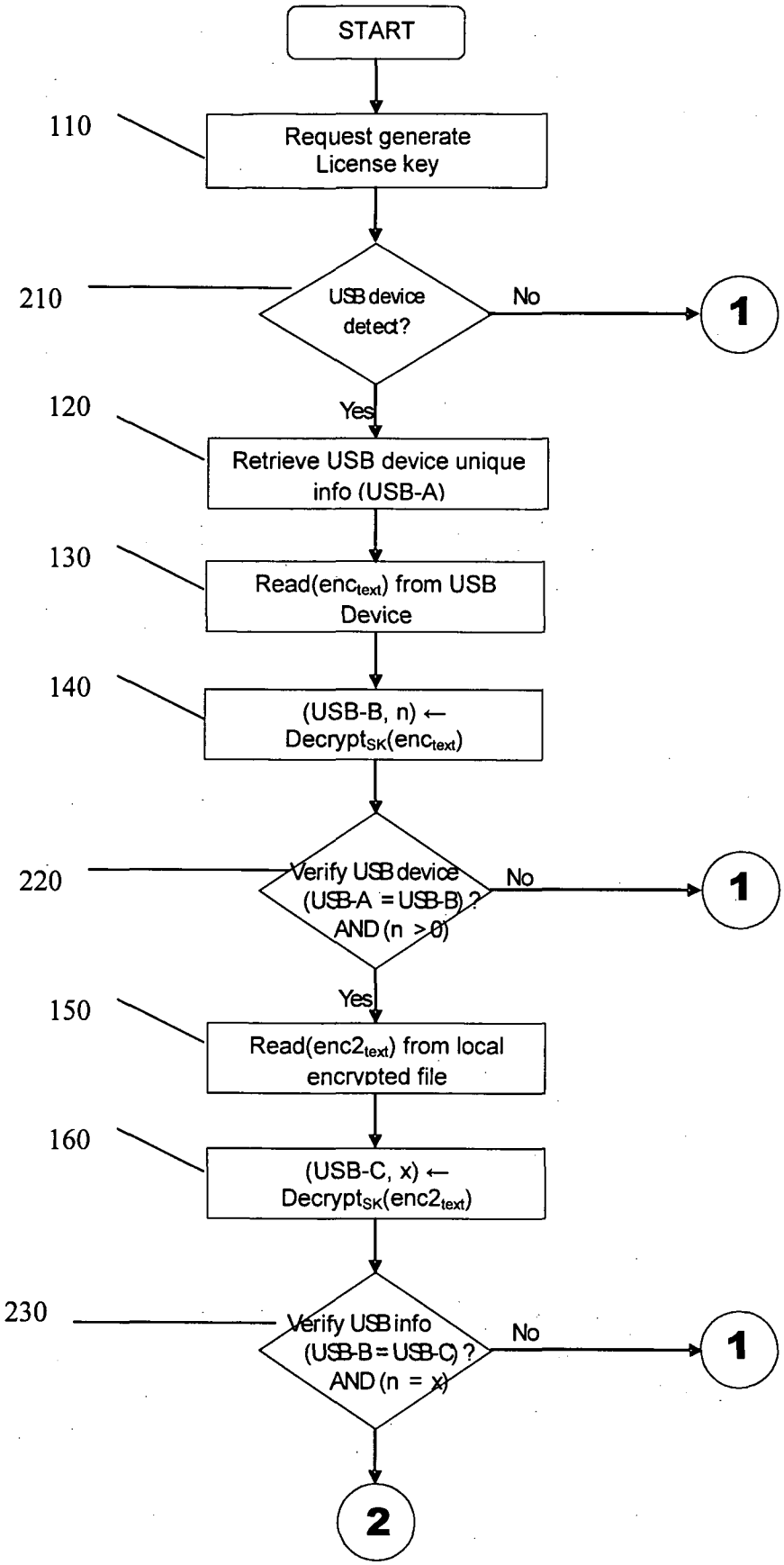
START
110
Request generate License key
210
USB device detect?
No
1
Yes
120
Retrieve USB device unique info (USB-A)
130
Read(enc$_{text}$) from USB Device
140
(USB-B, n) ← Decrypt$_{SK}$(enc$_{text}$)
220
Verify USB device (USB-A = USB-B) ? AND (n > 0)
No
1
Yes
150
Read(enc2$_{text}$) from local encrypted file
160
(USB-C, x) ← Decrypt$_{SK}$(enc2$_{text}$)
230
Verify USB info (USB-B = USB-C) ? AND (n = x)
No
1
2

Figure 1

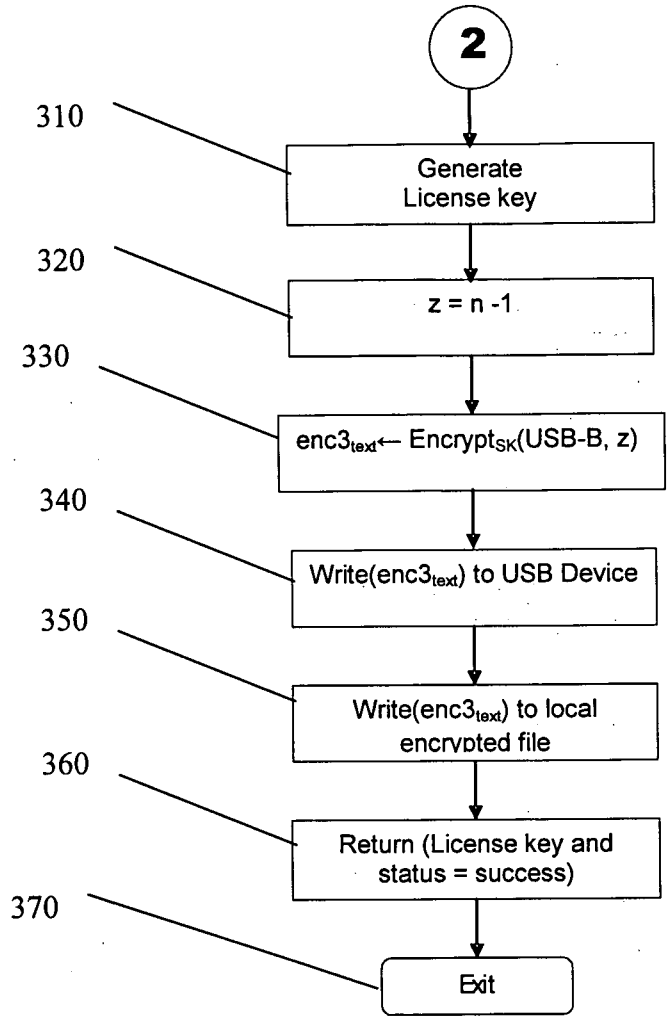
2
310
Generate License key
320
z = n -1
330
enc3text← EncryptSK(USB-B, z)
340
Write(enc3text) to USB Device
350
Write(enc3text) to local encrypted file
360
Return (License key and status = success)
370
Exit

Figure 2

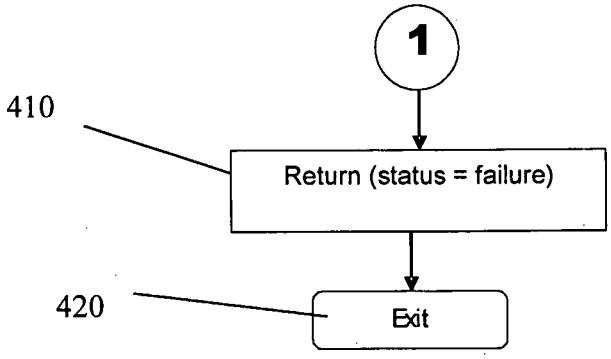

Figure 3

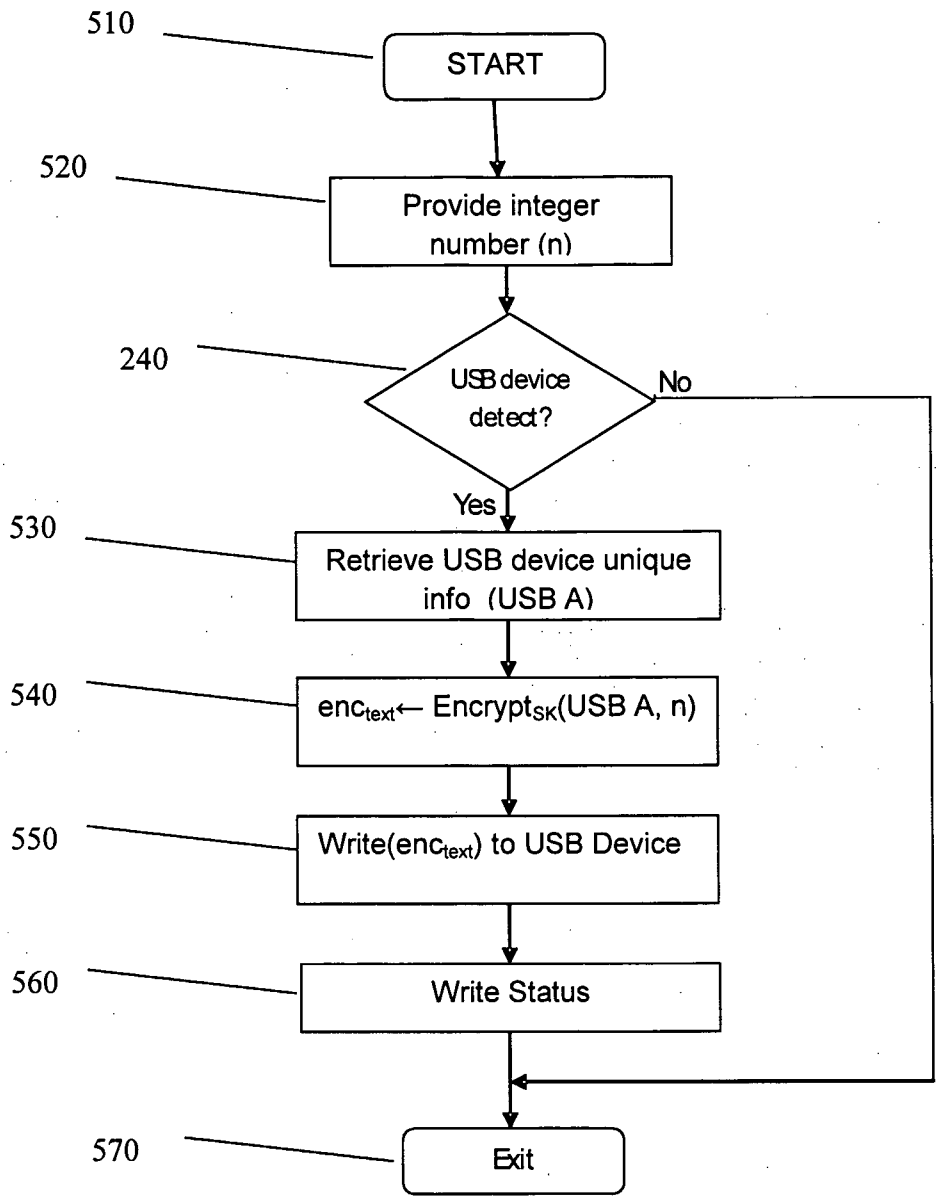
510
START
520
Provide integer number (n)
240
USB device detect?
No
Yes
530
Retrieve USB device unique info (USB A)
540
enc_{text} ← $Encrypt_{SK}$(USB A, n)
550
Write(enc_{text}) to USB Device
560
Write Status
570
Exit

Figure 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/MY2011/000069

A. CLASSIFICATION OF SUBJECT MATTER

Int. Cl.

G06F 21/00 (2006.01) ***H04L 9/08*** (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)
WPI, EPODOC, Google Patent, E-Sp@ce & keywords (licen+, key, token, comparison) & similar terms

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 7805616 B1 (MOHAMMED et al.) 28 September 2010 (See Whole Document)	1-5
A	US 2004/0098348 A1 (KAWASAKI et al.) 20 May 2004 (See Whole Document)	1-5

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

* Special categories of cited documents:

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
28 September 2011	29/09/2011
Name and mailing address of the ISA/AU AUSTRALIAN PATENT OFFICE PO BOX 200, WODEN ACT 2606, AUSTRALIA E-mail address: pct@ipaustralia.gov.au Facsimile No. +61 2 6283 7999	Authorized officer **KANWAL PAHWA** AUSTRALIAN PATENT OFFICE (ISO 9001 Quality Certified Service) Telephone No : +61 2 6283 2644

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/MY2011/000069

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	7805616	NONE	
US	2004098348	JP	2004110646

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

END OF ANNEX