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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 June 2015 (11.06.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/084144 A1

(51) International Patent Classification:
G06F 21/57 (2013.01) *H04L 29/06* (2006.01)

(21) International Application Number:
PCT/MY2014/000158

(22) International Filing Date:
4 June 2014 (04.06.2014)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
PI 2013004370 4 December 2013 (04.12.2013) MY

(71) Applicant: **MIMOS BERHAD** [MY/MY]; Technology Park Malaysia, 57000 Kuala Lumpur (MY).

(72) Inventors: **ABD AZIZ, Norazah**; c/o MIMOS Berhad, Technology Park Malaysia, 57000 Kuala Lumpur (MY). **BHAGYALAXMI, Aakula**; c/o MIMOS Berhad, Technology Park Malaysia, 57000 Kuala Lumpur (MY).

(74) Agent: **MIRANDAH, Patrick**; Mirandah Asia (Malaysia) sdn bhd, Suite 3B-19-3, Plaza Sentral, Jalan Stesen Sentral 5, 50470 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, BN, 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, IR, IS, JP, KE, KG, 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, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: A SYSTEM AND METHOD TO SECURE VIRTUAL MACHINE IMAGES IN CLOUD COMPUTING

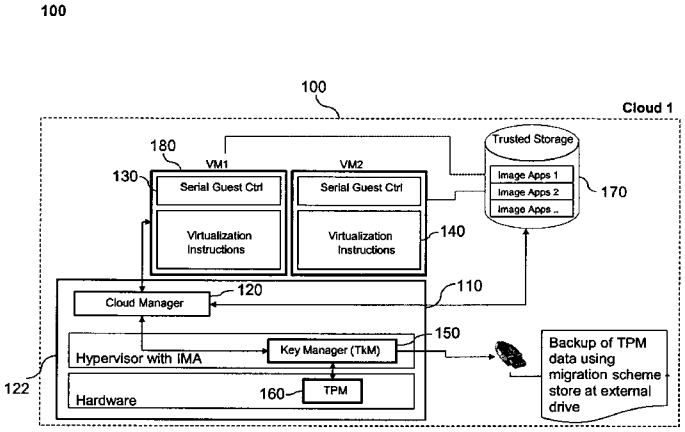

FIG. 1.0

(57) Abstract: The system (100) of the present invention to secure Virtual Machine images in cloud computing comprising at least one hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160); at least one Cloud Manager (CM) module (120) configured with serial communication function; at least one trusted storage server (170) storing modified Virtual Machine images with sealed key indexed by Virtual Machine Universally Unique Identifier (UUID); and at least one Serial Guest Control interface (130) embedded in kernel module configured with serial communication function and interface to said Cloud Manager (CM) module (120). The general methodology of the present invention comprises steps of configuring a server with at least one Cloud Manager (CM) module and at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160) by creating new Virtual Machines in the cloud (202); generating Trusted Platform Module (TPM) Key for Virtual Machine (206); installing and compiling Virtual Machines with new module containing encrypted static object of kernel module with said symmetric key (208); sealing said symmetric key of the Virtual Machine associated with Trusted Platform Module (TPM) with Virtual Machine Universally Unique Identifier (UUID) (210, 212); storing said sealed key and modified Virtual Machine images indexed

[Continued on next page]

WO 2015/084144 A1

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *of inventorship (Rule 4.17(iv))*

Published:

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

with Virtual Machine Universally Unique Identifier (UUID) into a trusted storage server (214); and accessing said Virtual Machines by decrypting said static object of kernel module using stored unseal symmetric key during booting process (216). The distinctiveness lies in the utilization of embedded new module comprising static object encryption module and built-in serial communication in the kernel of Virtual Machine (VM) images to provide a system and method to protect Virtual Machine (VM) images from running in different cloud providers.

A SYSTEM AND METHOD TO SECURE VIRTUAL MACHINE IMAGES IN CLOUD COMPUTING

FIELD OF INVENTION

The present invention relates to a system and method to secure virtual machine images in cloud computing. In particular, the invention relates to a system and method which provides a system and method to protect virtual machine images from running in different cloud providers by utilizing new embedded module.

BACKGROUND ART

Cloud computing is a technology that utilizes the internet and central remote servers to maintain data and applications which further allows consumers and businesses to use applications without installation of the application and having access to their personal files at any computer. Cloud computing provides efficient computing with centralized data storage, processing and bandwidth.

Despite the cloud's huge potential in reducing costs and improving productivity, security experts opines that security problems could restrain wide adoption in the cloud model. There are many open questions in regards to security of cloud computing which cannot be managed in traditional ways. Examples of the scenario are as listed below:

1) Cloud virtualization technologies comprise virtual machine (VM) that can run on different cloud providers. Different organizations who lease VMs from the same cloud providers may compromise the virtualization layer of the cloud provider with the intention to obtain information regarding the other customer's VMs by extracting the unauthorized VM images.
2) Different billing rates by different hypervisor/cloud providers encourage customers of the hypervisor/one cloud provider to take VM images and run them on another hypervisor/cloud provider which offer a lower cost. Consequently, the first cloud provider will lose some revenue.

The issue of the currently available deployed system drives to need of protecting virtual machine (VM) images from running in different hypervisor/cloud providers.

One example of currently available method for generating a dedicated virtual machine image is disclosed in United States Patent Application Publication No. US 2009/0172781 A1 entitled "Trusted Virtual Machine as a Client" (hereinafter referred to as the US' 781 Publication). In the US' 781 Publication, storage of the encrypted virtual machine image is provided in a portable computing device as compared to the present application which stores modified virtual machine image and sealed key indexing with Universally Unique Identifier (UUID) in a trusted storage server. Further, in the US' 781 Publication, remote attestation mechanism is utilize to verify integrity of the launched virtual machine and hypervisor through service provider; the service provider determines the level of trust for each virtual machines as compared to the present application which utilizes Integrity Measurement Architecture (IMA) to protect hypervisor integrity, launched virtual machine and modified kernel module to encrypt some of the static kernel object in virtual machine images for verification process during booting-up.

Another example which describes techniques for securely booting and executing a virtual machine (VM) image in an untrusted cloud infrastructure is disclosed in Untied States Patent Publication No. US 2011/0302400 A1 entitled "Secure Virtual Machine Bootstrap in Untrusted Cloud Infrastructures" (hereinafter referred to as the US' 400 Publication). In the US' 400 Publication, secure booting of a virtual machine image is provided in an untrusted cloud infrastructure. A trust anchor provision with public key pair that allows the multi-core CPU to authenticate itself which further securely booting and executing a virtual machine image in an untrusted cloud infrastructure is disclosed in the US' 400 Publication. In the US' 400 Publication, virtual machine image is encrypted by utilizing the key wrapped by the public key of service provider as compared to the present application which provides the kernel module embedded in virtual machine that is encrypted with key and is being sealed in TPM.

Another mechanism which provides secure, flexible and transparent security architecture for virtual disk images is disclosed in an IEEE paper (2008 IEEE) entitled "Secure Virtual Disk Images for Grid Computing" by Carl Gebhardt and Allan Tomlinson. In the said paper, virtual disks are secured by checking integrity using hash value of the whole image through driver. Splitting method is provided wherein the images are split into small chunks of a fixed length to avoid a constant rehashing of the virtual images as compared to the present invention which provides new kernel module with encryption key being embedded

in virtual machines. Further, the said IEEE paper provides metafile to store the encryption key of the chunks and allow a restrictive rule-set to be imposed by the virtual disk image owner. It does not provide for a new kernel module embedded in virtual machine, encrypted with key and sealed in Trusted Platform Module (TPM) as provided in the present invention.

SUMMARY OF INVENTION

The present invention relates to a system and method to determine version of deployed package. In particular, the invention relates to a system and method which provides for a version of a deployed package to be determined and verified against release packages stored in a versioning repository.

One aspect of the invention provides a system (100) to secure Virtual Machine images in cloud computing. The system comprising at least one hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160); at least one Cloud Manager (CM) module (120) configured with serial communication function; at least one trusted storage server (170) storing modified Virtual Machine images with sealed key indexed by Virtual Machine Universally Unique Identifier (UUID); at least one Serial Guest Control interface (130) embedded in kernel module configured with serial communication function and interface to said Cloud Manager (CM) module (120). The at least one hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160) further comprises static object encryption module which utilizes Trusted Platform Module (TPM) seal functionalities to retrieve key for encryption and decryption.

Another aspect of the invention provides a method (200) to secure Virtual Machine images in cloud computing. The method comprising steps of configuring a server with at least one Cloud Manager (CM) module and at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160) by creating new Virtual Machines in the cloud (202); generating Trusted Platform Module (TPM) Key for Virtual Machine (206); installing and compiling Virtual Machines with new module containing encrypted static object of kernel module with said symmetric key (208); sealing said symmetric key of the Virtual Machine associated with Trusted Platform Module (TPM) with Virtual Machine Universally Unique Identifier (UUID) (210, 212); storing said sealed key and modified Virtual Machine images indexed with Virtual Machine Universally Unique Identifier (UUID) into a trusted storage server (214); and accessing said Virtual Machines by decrypting said static object of kernel module using

stored unseal symmetric key during booting process (216). The step for accessing said Virtual Machines by decrypting said static object of kernel module using stored unseal symmetric key during booting process (216) further comprises steps of communicating with trusted storage server to access Virtual Machine image and sealed Trusted Platform Module (TPM) key (402); establishing Virtual Machine image (404); receiving request from Virtual Machine during booting process requesting symmetric key from Serial Guest Control (SGC) of said Virtual Machine through serial communication (406); forwarding said request by Cloud Manager (CM) module to Trusted Platform Module (TPM) Key Manager (TkM) module to unseal said sealed key (408); and providing access to user by decrypting said static object (410a, 410b).

Yet another aspect of the invention relates to the step for configuring a server with at least one Cloud Manager (CM) module and at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160) by creating new Virtual Machines in the cloud (202). The said step further comprises steps of connecting to the at least one Trusted Platform Module (TPM) Key Manager (TkM) module by the at least one Cloud Manager (CM) module to obtain a symmetric key (302, 304); inserting a new module into Virtual Machine image module by the at least one Cloud Manager (CM) module to boot the Virtual Machine (306); and communicating with said Trusted Platform Module (TPM) Key Manager (TkM) module by said Cloud Manager (CM) module to seal said symmetric key and storing said modified Virtual Machine image with sealed key by indexing with said Virtual Machine Universally Unique Identifier (UUID) into trusted storage server. The step for connecting to the at least one Trusted Platform Module (TPM) Key Manager (TkM) module by the at least one Cloud Manager (CM) module to obtain a symmetric key (302, 304) further comprises connecting said Trusted Platform Module (TPM) Key Manager (TkM) module to Trusted Platform Module (160) to generate symmetric key.

Still another aspect of the invention provides for the step for inserting a new module into Virtual Machine image module by the at least one Cloud Manager (CM) module to boot the Virtual Machine (306) which further comprises steps of encrypting static object of kernel module with said symmetric key to boot the Virtual Machine; and compiling Virtual Machine with new module, sending the signal to the Cloud Manager (CM) module and shuts off.

A further aspect of the invention provides for the step for establishing Virtual Machine image. The said step further comprises steps of requesting Virtual Machine image and sealed key from trusted storage server (502); forwarding signal to Cloud Manager (CM) module through serial communication to enable said key to decrypt static object in new kernel module (504); requesting unseal Trusted Platform Module (TPM) Key (506) by communicating to Trusted Platform Module (TPM) to unseal said key and forwarding said symmetric key to Cloud Manager (CM) module (508); forwarding respond signal by Cloud Manager (CM) module through serial communication with Trusted Platform Module (TPM) symmetric key (510); and decrypting static object of new kernel module using symmetric key (512); opening said connection for access upon valid decryption of static object of kernel module; else halting said Virtual Machine by signaling from new kernel module decryption of static object is invalid.

The present invention consists of features and a combination of parts hereinafter fully described and illustrated in the accompanying drawings, it being understood that various changes in the details may be made without departing from the scope of the invention or sacrificing any of the advantages of the present invention.

BRIEF DESCRIPTION OF ACCOMPANYING DRAWINGS

To further clarify various aspects of some embodiments of the present invention, a more particular description of the invention will be rendered by references to specific embodiments thereof, which are illustrated in the appended drawings. It is appreciated that these drawings depict only typical embodiments of the invention and are therefore not to be considered limiting of its scope. The invention will be described and explained with additional specificity and detail through the accompanying drawings in which:

FIG. 1.0 illustrates the general architecture of the system of the present invention.

FIG. 2.0 is a flowchart illustrating the general methodology of the present invention to secure virtual machine images in cloud computing.

FIG. 3.0 is a flow diagram which illustrates the step for creating new Virtual Machines in the cloud.

FIG. 4.0 is a flow diagram which illustrates the step for accessing said Virtual Machines by decrypting said static object of kernel module using stored unseal symmetric key during booting process.

FIG. 5.0 is a flowchart illustrating the steps for establishing Virtual Machine image.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention relates to a system and method to secure virtual machine images in cloud computing. In particular, the invention relates to a system and method which provides a system and method to protect virtual machine images from running in different cloud providers by utilizing new embedded module.

Hereinafter, this specification will describe the present invention according to the preferred embodiments. It is to be understood that limiting the description to the preferred embodiments of the invention is merely to facilitate discussion of the present invention and it is envisioned without departing from the scope of the appended claims.

Referring to FIG. 1.0, a system to secure virtual machine images in cloud computing is illustrated. As illustrated in FIG. 1.0, the system (100) comprising a hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with a Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with a Trusted Platform Module (160); a Cloud Manager (CM) module (120) configured with serial communication function; a trusted storage server (170) storing modified Virtual Machine images with sealed key indexed by Virtual Machine Universally Unique Identifier (UUID); a Serial Guest Control interface (130) embedded in kernel module configured with serial communication function and interface to said Cloud Manager (CM) module (120). The hypervisor with Integrity Measurement Architecture (IMA) (122) is embedded with the Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with a Trusted Platform Module (160) further comprises static object encryption module which utilizes Trusted Platform Module (TPM) seal functionalities to retrieve key for encryption and decryption. The sealed key as described is generated by the standard of trusted platform module (TPM) sealing process. The hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with the Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with the Trusted Platform Module (160) generates a Trusted Platform Module (TPM) Key; said TPM Key is a symmetric key.

A general method (200) of an embodiment of the invention is illustrated in FIG. 2.0 wherein the method (200) to secure virtual machine images in cloud computing further comprising steps of configuring a server with at least one Cloud Manager (CM) module

and at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160) by creating new Virtual Machines in the cloud (202); generating Trusted Platform Module (TPM) Key for Virtual Machine (206); installing and compiling Virtual Machines with new module containing encrypted static object of kernel module with said symmetric key (208); sealing said symmetric key of the Virtual Machine associated with Trusted Platform Module (TPM) with Virtual Machine Universally Unique Identifier (UUID) (210, 212); storing said sealed key and modified Virtual Machine images indexed with Virtual Machine Universally Unique Identifier (UUID) into a trusted storage server (214); and accessing said Virtual Machines by decrypting said static object of kernel module using stored unseal symmetric key during booting process (216). A detailed description of the steps to create new Virtual Machines in the cloud is illustrated in FIG 3.0. As illustrated in FIG. 3.0, Cloud Manager (CM) module connects to the Trusted Platform Module (TPM) Key Manager (TkM) module to obtain a symmetric key (302, 304). A symmetric key is generated by connecting said Trusted Platform Module (TPM) Key Manager (TkM) module to Trusted Platform Module. The Cloud Manager (CM) module boots the Virtual Machine by inserting a new module into Virtual Machine image module (306) by encrypting static object of kernel module with said symmetric key to boot the Virtual Machine; and compiling Virtual Machine with new module, sending the signal to the Cloud Manager (CM) module and shuts off. Thereafter, the Cloud Manager (CM) module communicates with the Trusted Platform Module (TPM) Key Manager (TkM) module to seal said symmetric key and store said modified Virtual Machine image with sealed key by indexing with said Virtual Machine Universally Unique Identifier (UUID) into trusted storage server.

Referring to FIG. 4.0, the steps for accessing Virtual Machines by decrypting static object of kernel module using stored unseal symmetric key during booting process is illustrated. As illustrated in FIG. 4.0, Cloud Manager (CM) module communicates with trusted storage server to access Virtual Machine image and sealed Trusted Platform Module (TPM) key (402). Thereafter, the Cloud Manager (CM) module will establish the Virtual Machine image (404). During the booting process, the Virtual Machine request for symmetric key from Serial Guest Control (SGC) of said Virtual Machine through serial communication (406). The Cloud Manager (CM) module forwards said request to Trusted Platform Module (TPM) Key Manager (TkM) module to unseal said sealed key (408) and the Cloud Manager (CM) module will reforward the key to the Serial Guest Control (SGC). The

Virtual Machine (VM) will decrypt the new module static object and provide access to user (410a, 410b).

The present invention manages access of Virtual Machine (VM) images in a secure manner by preventing Virtual Machine (VMs) from running on other cloud providers and hypervisor. The distinctiveness of the present invention lies in the utilization of embedded new module comprising static object encryption module and built-in serial communication in the kernel of Virtual Machine (VM) images. The said encryption utilizes Trusted Platform Module (TPM) seal functionalities while the serial communication is for communication between the running Virtual Machine (VM) during booting process with Cloud Manager (CM) in order to retrieve the key for decryption process.

Unless the context requires otherwise or specifically stated to the contrary, integers, steps or elements of the invention recited herein as singular integers, steps or elements clearly encompass both singular and plural forms of the recited integers, steps or elements.

Throughout this specification, unless the context requires otherwise, the word "comprise", or variations such as "comprises" or "comprising", will be understood to imply the inclusion of a stated step or element or integer or group of steps or elements or integers, but not the exclusion of any other step or element or integer or group of steps, elements or integers. Thus, in the context of this specification, the term "comprising" is used in an inclusive sense and thus should be understood as meaning "including principally, but not necessarily solely".

It will be appreciated that the foregoing description has been given by way of illustrative example of the invention and that all such modifications and variations thereto as would be apparent to persons of skill in the art are deemed to fall within the broad scope and ambit of the invention as herein set forth.

CLAIMS

1. A system (100) to secure Virtual Machine images in cloud computing comprising:

   at least one hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160);

   at least one Cloud Manager (CM) module (120) configured with serial communication function;

   at least one trusted storage server (170) storing modified Virtual Machine images with sealed key indexed by Virtual Machine Universally Unique Identifier (UUID);

   at least one Serial Guest Control interface (130) embedded in kernel module configured with serial communication function and interface to said Cloud Manager (CM) module (120)

   characterized in that

   the at least one hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160) further comprises static object encryption module which utilizes Trusted Platform Module (TPM) seal functionalities to retrieve key for encryption and decryption.

2. A system (100) according to Claim 1, wherein the at least one hypervisor with Integrity Measurement Architecture (IMA) (122) embedded with at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform Module (160) generates Trusted Platform Module (TPM) Key; said TPM Key is a symmetric key.

3. A method (200) to secure Virtual Machine images in cloud computing comprising steps of:

   configuring a server with at least one Cloud Manager (CM) module and at least one Trusted Platform Module (TPM) Key Manager (TkM) module

(150) associated with at least one Trusted Platform Module (160) by creating new Virtual Machines in the cloud (202);

generating Trusted Platform Module (TPM) Key for Virtual Machine (206);

installing and compiling Virtual Machines with new module containing encrypted static object of kernel module with said symmetric key (208);

sealing said symmetric key of the Virtual Machine associated with Trusted Platform Module (TPM) with Virtual Machine Universally Unique Identifier (UUID) (210, 212);

storing said sealed key and modified Virtual Machine images indexed with Virtual Machine Universally Unique Identifier (UUID) into a trusted storage server (214); and

accessing said Virtual Machines by decrypting said static object of kernel module using stored unseal symmetric key during booting process (216) characterized in that

accessing said Virtual Machines by decrypting said static object of kernel module using stored unseal symmetric key during booting process (216) further comprises steps of:

communicating with trusted storage server to access Virtual Machine image and sealed Trusted Platform Module (TPM) key (402);

establishing Virtual Machine image (404);

receiving request from Virtual Machine during booting process requesting symmetric key from Serial Guest Control (SGC) of said Virtual Machine through serial communication (406);

forwarding said request by Cloud Manager (CM) module to Trusted Platform Module (TPM) Key Manager (TkM) module to unseal said sealed key (408); and

providing access to user by decrypting said static object (410a, 410b).

4. A method according to Claim 3, wherein configuring a server with at least one Cloud Manager (CM) module and at least one Trusted Platform Module (TPM) Key Manager (TkM) module (150) associated with at least one Trusted Platform

Module (160) by creating new Virtual Machines in the cloud (202) further comprises steps of:

connecting to the at least one Trusted Platform Module (TPM) Key Manager (TkM) module by the at least one Cloud Manager (CM) module to obtain a symmetric key (302, 304);

inserting a new module into Virtual Machine image module by the at least one Cloud Manager (CM) module to boot the Virtual Machine (306); and

communicating with said Trusted Platform Module (TPM) Key Manager (TkM) module by said Cloud Manager (CM) module to seal said symmetric key and storing said modified Virtual Machine image with sealed key by indexing with said Virtual Machine Universally Unique Identifier (UUID) into trusted storage server.

5. A method according to Claim 4, wherein connecting to the at least one Trusted Platform Module (TPM) Key Manager (TkM) module by the at least one Cloud Manager (CM) module to obtain a symmetric key (302, 304) further comprises connecting said Trusted Platform Module (TPM) Key Manager (TkM) module to Trusted Platform Module (160) to generate symmetric key.

6. A method according to Claim 4, wherein inserting a new module into Virtual Machine image module by the at least one Cloud Manager (CM) module to boot the Virtual Machine (306) further comprises steps of:

encrypting static object of kernel module with said symmetric key to boot the Virtual Machine; and

compiling Virtual Machine with new module , sending the signal to the Cloud Manager (CM) module and shuts off.

7. A method (500) according to Claim 3, wherein establishing Virtual Machine image further comprising steps of:

requesting Virtual Machine image and sealed key from trusted storage server (502);

forwarding signal to Cloud Manager (CM) module through serial communication to enable said key to decrypt static object in new kernel module (504);

requesting unseal Trusted Platform Module (TPM) Key (506) by communicating to Trusted Platform Module (TPM) to unseal said key and forwarding said symmetric key to Cloud Manager (CM) module (508);

forwarding respond signal by Cloud Manager (CM) module through serial communication with Trusted Platform Module (TPM) symmetric key (510); and

decrypting static object of new kernel module using symmetric key (512);

opening said connection for access upon valid decryption of static object of kernel module; else halting said Virtual Machine by signaling from new kernel module decryption of static object is invalid.

8. A method according to Claim 3, wherein said new module containing encrypted static object of kernel module with said symmetric key is embedded with Serial Guest Control (SGC).

100

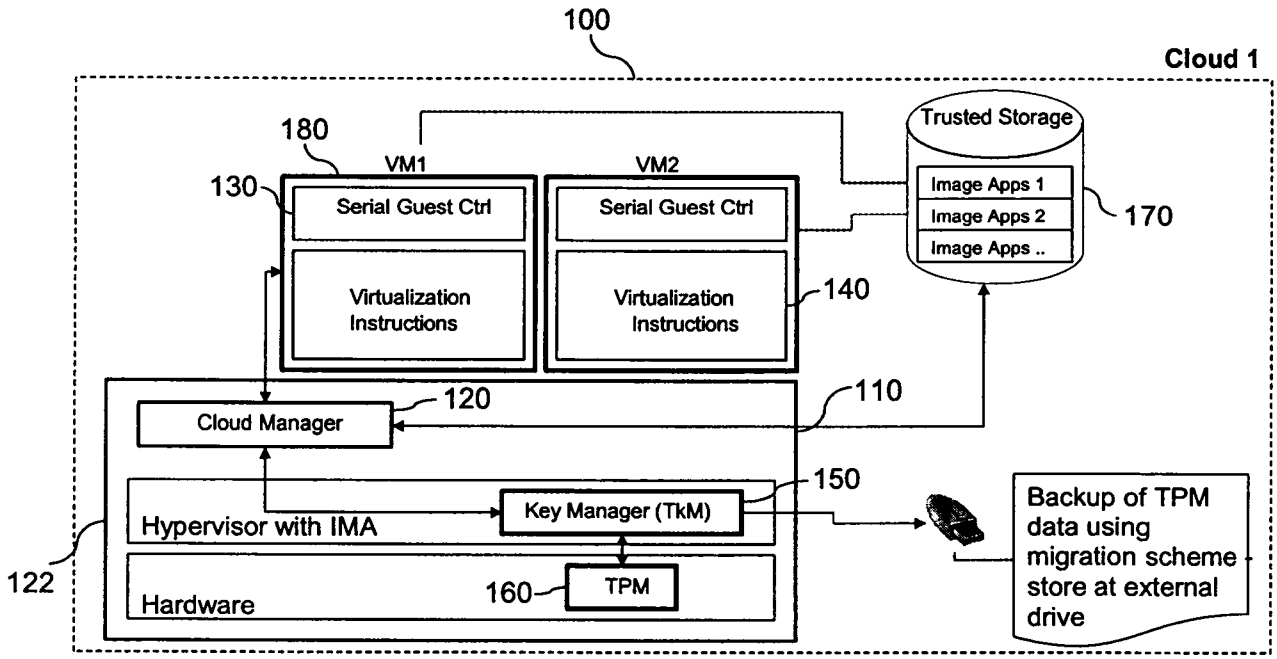
100
Cloud 1
180
VM1
VM2
130
Serial Guest Ctrl
Serial Guest Ctrl
Virtualization Instructions
Virtualization Instructions
140
Trusted Storage
Image Apps 1
Image Apps 2
Image Apps ..
170
120
Cloud Manager
110
150
Key Manager (TkM)
Hypervisor with IMA
122
Hardware
160
TPM
Backup of TPM data using migration scheme store at external drive

FIG. 1.0

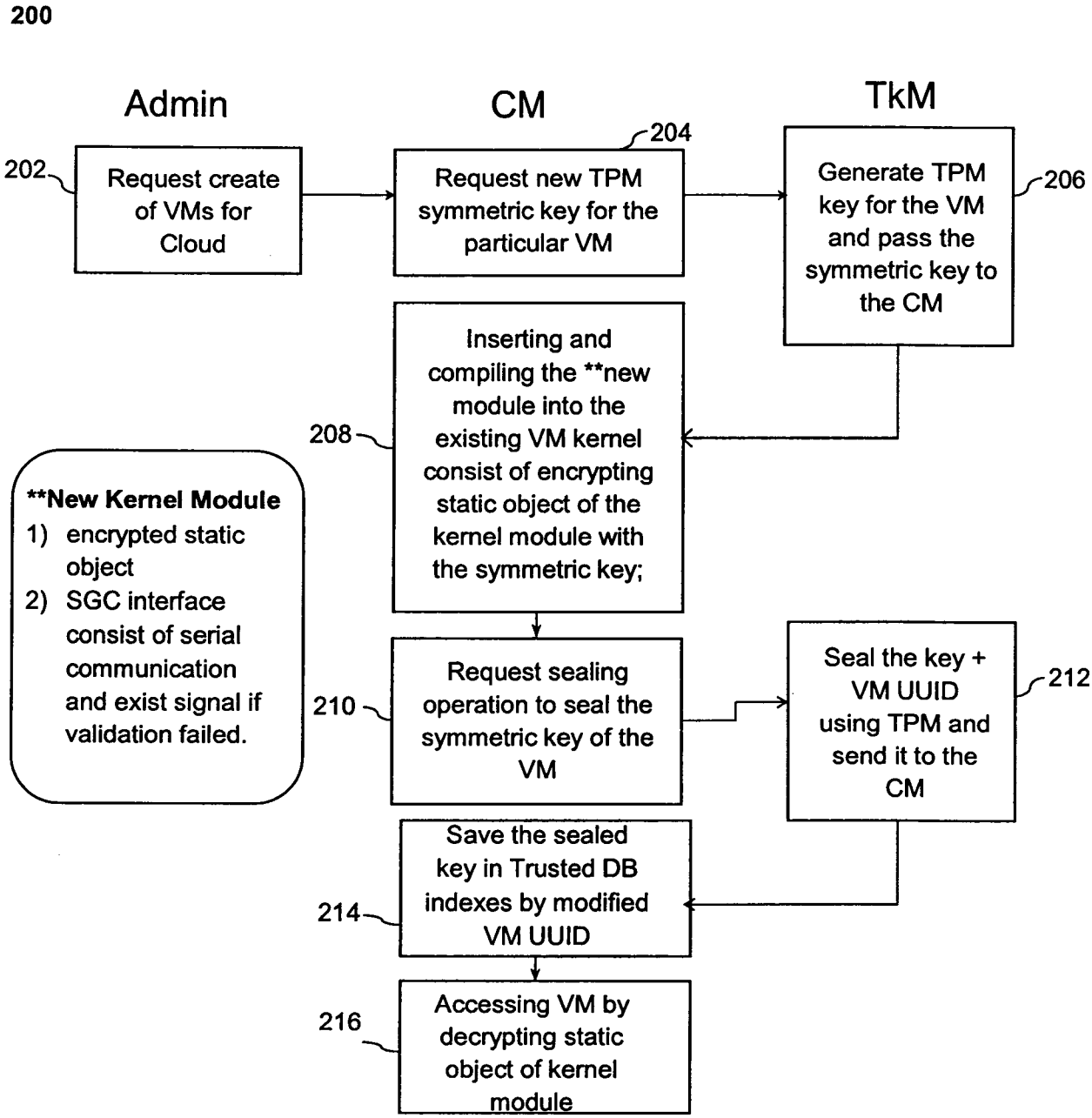
200
Admin
CM
TkM
202
Request create of VMs for Cloud
204
Request new TPM symmetric key for the particular VM
Generate TPM key for the VM and pass the symmetric key to the CM
206
Inserting and compiling the **new module into the existing VM kernel consist of encrypting static object of the kernel module with the symmetric key;
208
**New Kernel Module
1) encrypted static object
2) SGC interface consist of serial communication and exist signal if validation failed.
Request sealing operation to seal the symmetric key of the VM
210
Seal the key + VM UUID using TPM and send it to the CM
212
Save the sealed key in Trusted DB indexes by modified VM UUID
214
Accessing VM by decrypting static object of kernel module
216

FIG. 2.0

300

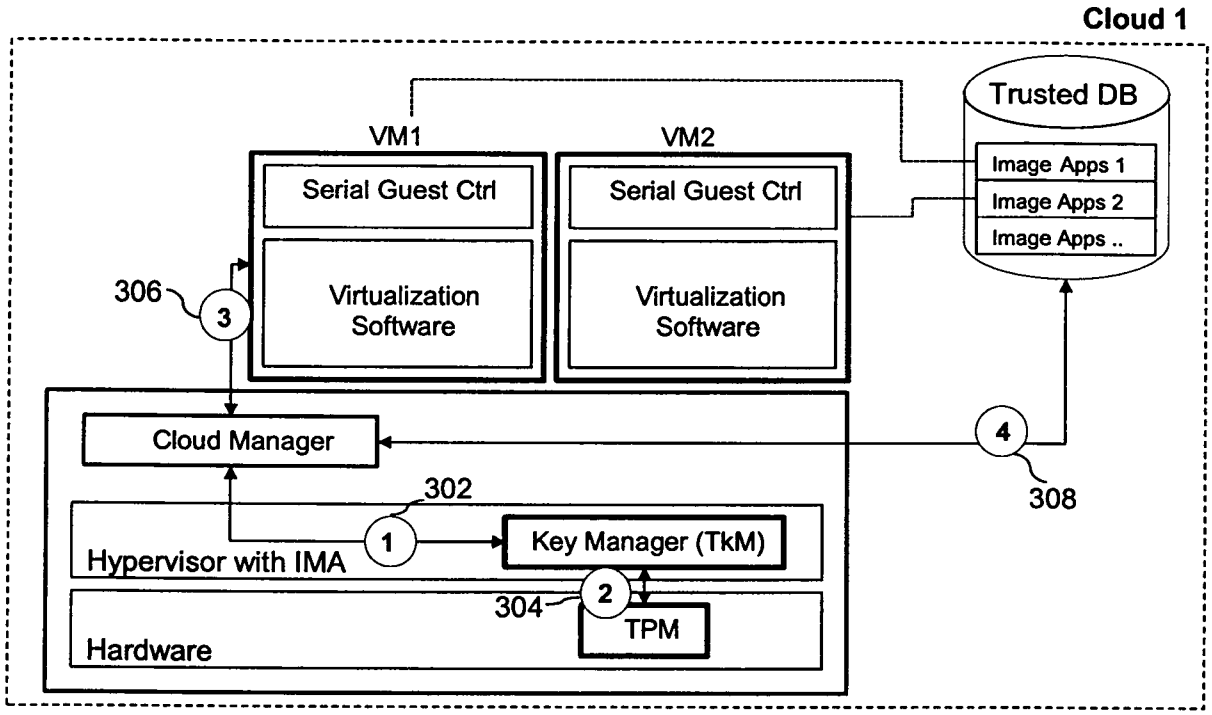
Cloud 1
Trusted DB
Image Apps 1
Image Apps 2
Image Apps ..
VM1
VM2
Serial Guest Ctrl
Serial Guest Ctrl
Virtualization Software
Virtualization Software
306
3
Cloud Manager
4
308
302
1
Key Manager (TkM)
Hypervisor with IMA
2
304
TPM
Hardware

FIG. 3.0

400

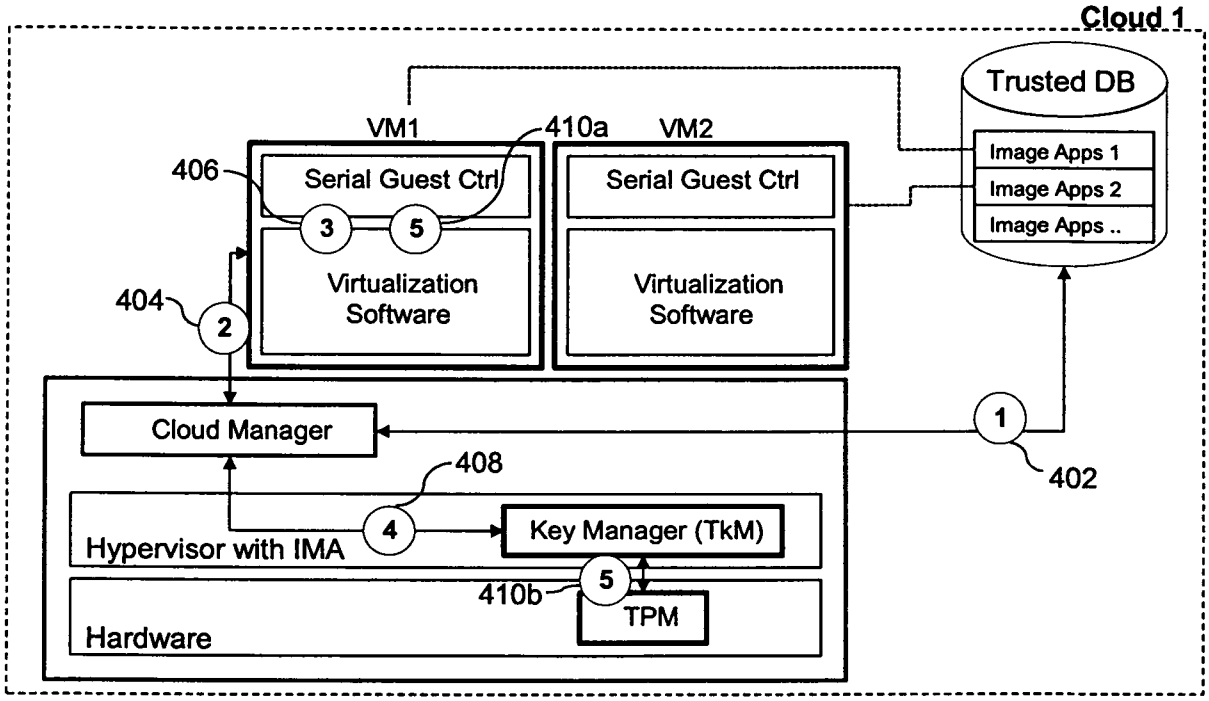
Cloud 1
Trusted DB
Image Apps 1
Image Apps 2
Image Apps ..
VM1
410a
VM2
406
Serial Guest Ctrl
Serial Guest Ctrl
3
5
Virtualization Software
Virtualization Software
404
2
Cloud Manager
1
402
408
4
Key Manager (TkM)
Hypervisor with IMA
5
410b
TPM
Hardware

FIG. 4.0

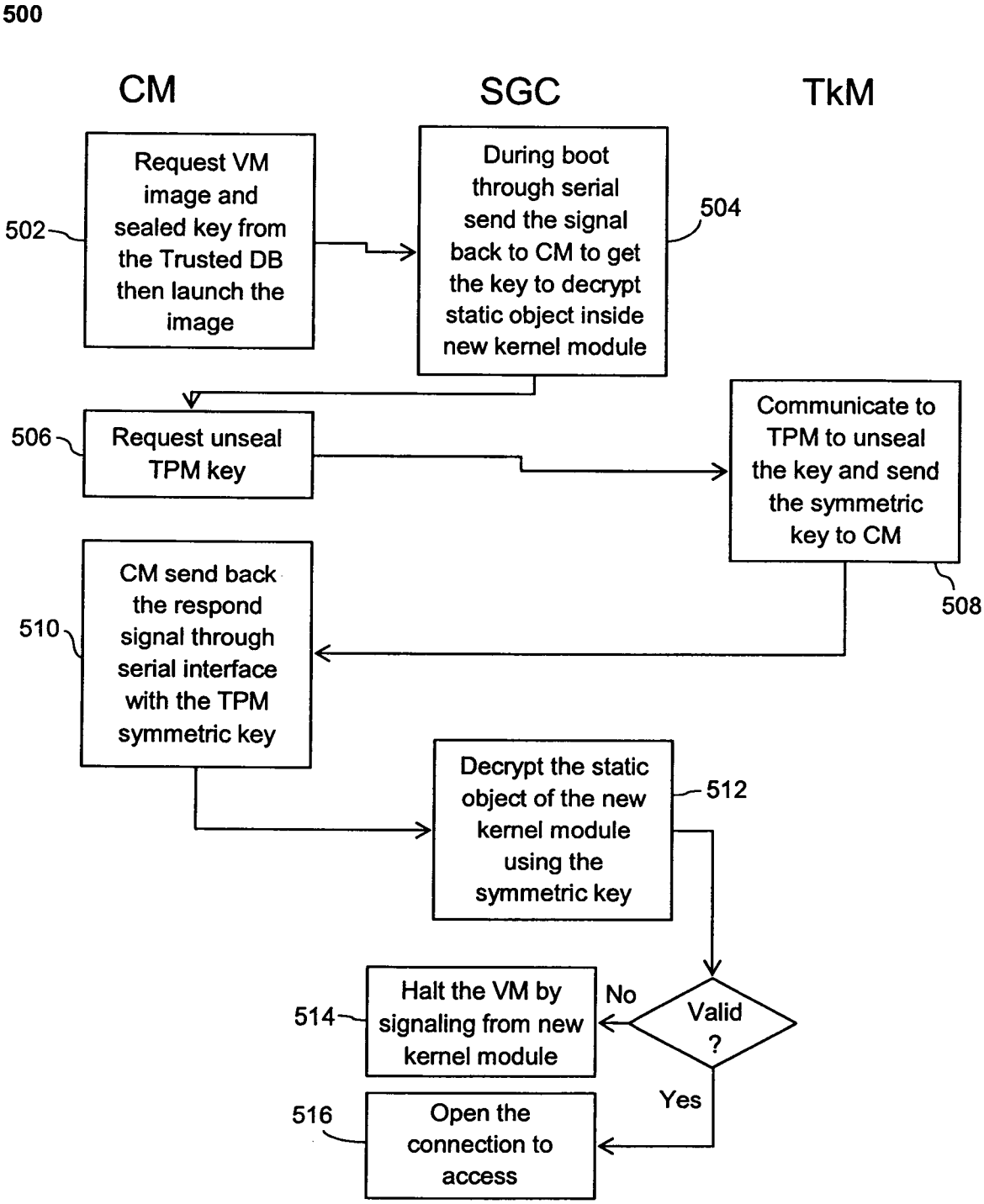
500
CM
SGC
TkM
Request VM image and sealed key from the Trusted DB then launch the image
502
During boot through serial send the signal back to CM to get the key to decrypt static object inside new kernel module
504
Request unseal TPM key
506
Communicate to TPM to unseal the key and send the symmetric key to CM
508
CM send back the respond signal through serial interface with the TPM symmetric key
510
Decrypt the static object of the new kernel module using the symmetric key
512
Valid ?
No
Yes
Halt the VM by signaling from new kernel module
514
Open the connection to access
516

FIG. 5.0

INTERNATIONAL SEARCH REPORT

International application No
PCT/MY2014/000158

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F21/57 H04L29/06
ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F H04L

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2011/302400 A1 (MAINO FABIO R [US] ET AL) 8 December 2011 (2011-12-08) cited in the application paragraph [0026] - paragraph [0027]; figure 1A paragraph [0032] paragraph [0036] - paragraph [0042]; figure 3 paragraph [0048] - paragraph [0051]; figure 5	1-8
X	US 2008/244569 A1 (CHALLENER DAVID CARROLL [US] ET AL) 2 October 2008 (2008-10-02) paragraph [0006] - paragraph [0008] paragraph [0032] - paragraph [0037]; figures 3,4 sentence 39, paragraph 38; figure 5	1-8

-/--

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

[X] See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
14 January 2015	20/01/2015

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

2

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/MY2014/000158

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 2011/116459 A1 (ENOMALY INC [CA]; LIE DAVID [CA]; COHEN REUVEN [CA]; REINER RICHARD [C) 29 September 2011 (2011-09-29) paragraph [0007] paragraph [0034] - paragraph [0035]	1-8

2

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/MY2014/000158

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2011302400 A1	08-12-2011	CN 103069428 A	24-04-2013
		EP 2577543 A1	10-04-2013
		US 2011302400 A1	08-12-2011
		WO 2011156261 A1	15-12-2011
US 2008244569 A1	02-10-2008	NONE	
WO 2011116459 A1	29-09-2011	CN 102947795 A	27-02-2013
		EP 2550621 A1	30-01-2013
		US 2013185812 A1	18-07-2013
		WO 2011116459 A1	29-09-2011