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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 April 2020 (23.04.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/077415 A1

(51) International Patent Classification:
G06F 21/30 (2013.01) *H04L 9/32* (2006.01)
H04L 9/08 (2006.01)

(21) International Application Number:
PCT/AU2019/051138

(22) International Filing Date:
18 October 2019 (18.10.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

2018903951	18 October 2018 (18.10.2018)	AU
2018903952	18 October 2018 (18.10.2018)	AU
2018903954	18 October 2018 (18.10.2018)	AU

(71) Applicant: **LOCKBOX TECHNOLOGIES PTY LTD** [AU/AU]; Level 1, 309-315 George Street, Sydney, New South Wales 2000 (AU).

(72) Inventors: **BROWN, Simon**. **ENGEL, Steven**.

(74) Agent: **WALLINGTON-DUMMER PATENT AND TRADE MARK ATTORNEYS**; GPO Box 3888, Sydney, New South Wales 2001 (AU).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: AUTHENTICATION SYSTEM

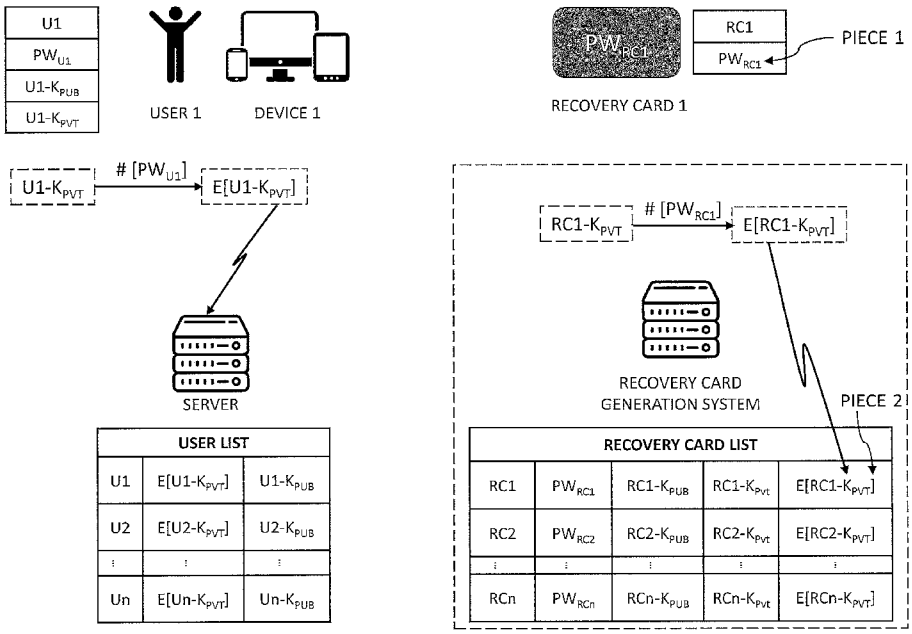


Figure 3A

(57) **Abstract:** A method, apparatus and systems are disclosed to control user access to digital items. Also disclosed is method, apparatus and systems which ensure secure storage and transmission of digital items within a system. Also disclosed is a password recovery methodology and apparatus.

[Continued on next page]

WO 2020/077415 A1

TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

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

Authentication System

TECHNICAL FIELD

[0001] The present invention relates to an authentication system and more particularly to a document storage and access system for digital items and more particularly but not exclusively such systems where it is desired to control user access to the digital items and to ensure secure storage and transmission of the digital items within the system.

BACKGROUND

[0002] A problem with centralized server systems which orchestrate access to documents is that they have a major security vulnerability if they store user passwords (or values derived from them) or the documents to which they orchestrate access. This is a major reason why servers are a target for hackers. Approaches to this issue include the following:

[0003] Enterprise digital rights management (e.g. Microsoft Office Rights Management) – however the systems do not have a storage platform which ensures that unprotected files cannot be uploaded. The systems do have a characteristic that protected data cannot be disclosed unless the user has been expressly authorised to see it.

[0004] Zero knowledge storage (e.g. SpiderOak). However, the systems provide only limited support for file sharing between users, and do not support partial item sharing (e.g. with or without annotation) and user delegation.

[0005] End to end messaging (e.g. Open Whisper Systems' Signal) systems. However, the systems do not have long-term storage model (rather than messaging) and typically cannot revoke a user's access once granted. They do not support user impersonation, nor support sharing outside of a closed system.

[0006] It is an object of the present invention to address or at least ameliorate some of the above disadvantages or provide a useful alternative.

NOTES

[0007] The term "comprising" (and grammatical variations thereof) is used in this specification in the inclusive sense

of “having” or “including”, and not in the exclusive sense of “consisting only of”.

[0008] The above discussion of the prior art in the Background of the invention, is not an admission that any information discussed therein is citable prior art or part of the common general knowledge of persons skilled in the art in any country.

SUMMARY OF INVENTION

DEFINITIONS

[0009] a “derivative” is a mathematical operation applied to a digital item so as to deterministically calculate a new result from the digital item. Examples include a hash of the digital item (including an iteratively repeated hash) or encryption of the digital item.

[0010] A “password” is a digital item associated with a user ID. Most usually the user ID is also a digital item such that the digital item is one digital item of a digital item hair which the proof of possession is required to effect authentication of the user ID on a digital device such that there is a level of trust that a login session on the digital device has been affected by was sponsored by that user represented by the user ID.

[0011] According to one broad form of the invention there is provided with a system for secure access control and transmission of digital items; the system including at least one server and at least one client device; the server including at least one server processor in communication with a server memory; the server memory storing code for execution by the server processor; the at least one client device including at least one device processor in communication with a device memory; the device memory storing code for execution by the device processor; the system further including a recovery card having its own set of recovery card credentials; the apparatus utilised to effect password replacement for a user using the at least one client device to communicate with the at least one server;
a user authenticated with the server using a user set of credentials including a user ID and user password and user private key and user public key;

the device authenticated with the server using a device set of credentials including a device ID and device password and device private key and device public key;
the system reliant on creation of a recovery card having its own set of credentials including a recovery card ID, a recovery card password, a recovery card private key and a recovery card public key;
the system associating a selected recovery card with the user on the server whereby subsequently the user may utilise the device and selected ones of the recovery card credentials to replace selected ones of the user credentials on the server.

[0012] According to another broad form of the invention there is provided a system for authentication of a user login request in order to permit an authenticated user session between the user and a server on a device of the user in a client/server system where the device and user are previously jointly authenticated with respect to the server; the system including at least one server and at least one client device; the server including at least one server processor in communication with a server memory; the server memory storing code for execution by the server processor; the at least one client device including at least one device processor in communication with a device memory; the device memory storing code for execution by the device processor; the user having at least for a given user login: a user ID, a user password, a user private key and a user public key;
the device having at least a device ID, a device public key and a device private key;
the server in a user enrolment event for the user storing the public key of the user and a derivative of the user private key of the user against the user ID;
the server in a device enrolment event for the device storing the device public key against the device ID;
said method comprising authenticating login of the user in a login event by the user to the server by the following steps:
provided the server has enrolled the user login;
provided the server has enrolled the device;
the device signals to the server commencement of the login event by the user;
the server supplies to the device the derivative of the user private key stored against the user ID and a randomly created authentication challenge;

the device performs a comparison operation to determine if the password of the user is the same as the password used to create the derivative to encrypt the user private key, and thereby decrypts the user private key from the derivative;
if the comparison operation is positive the device signals to the server that the user login request has been authenticated by providing a cryptographic response to the authentication challenge which requires possession of the user private key, whereupon the server treats the login event as successful such that the user ID is now authenticated for the user session on the server using the device.

[0013] According to another broad form of the invention there is provided a system for secure access control and transmission of digital items in a client server environment wherein at least one server communicates with a plurality of participating user entities by means of participating digital devices in the form of digital communications devices in order to provide secure access control and transmission services for digital items created by, originated by or brought into the environment by ones of the plurality of participating user entities;
the server and the device executing steps whereby:
for each participating user entity participating in the environment;
each participating user entity is authenticated as between the entity and the at least one server for that user entity in a participating user entity authentication step;
each participating digital device used by the participating user entity is authenticated as between the digital device and the at least one server for that user entity in a participating digital device authentication step;
and wherein each participating user entity has a participating user entity encryption key;
the steps further including:
a sending (source) user entity enabling secure access for a receiving (target) user entity of the plurality of user entities to a designated (unencrypted) digital item which is recorded as under the custody of the sending (source) user entity;
said sending (source) user entity encrypting the designated digital item or the user entity encrypting a key by which the designated digital item has been encrypted so as to produce an encrypted designated digital item or an encrypted digital item key preparatory to the sending user entity enabling secure

access to the designated digital item by the receiving (target) user;
said sending (source) user entity utilising a sending (source) user entity digital device which has been authenticated as between the server and the sending (source) user entity thereby to enable the designated receiving (target) user entity to decrypt the designated digital item;
said sending user entity having a sending user entity encryption key;
said designated receiving user entity having a designated receiving user entity encryption key;
access to the designated digital item effected by the sending user entity encrypting the designated digital item so as to produce the encrypted designated digital item by utilising the designated receiving user entity encryption key to encrypt the designated digital item or access to the designated digital item effected by the sending user entity encrypting the document key of the designated digital item so as to produce the encrypted document key of the designated digital item by utilising the designated receiving user entity encryption key to encrypt the document key of the designated digital item.

[0014] Accordingly in another broad form of the invention there is provided a password recovery method utilising a recovery card, said method comprising:
in a first card generation step
executed on an off-line device;
for each card the off-line device generates a card private key and
a card password and
a card ID and
a card public key (derived from the card private key);
the off-line device then encrypts the card private key using the card password so as to generate an encrypted card private key;
the off-line device then embosses and/or prints on the card so as to be readable by a user the card ID and the card password;
the system administrator may then upload to an online server (platform)
the card ID and
the encrypted card private key and
the card public key;
in a second user setup step the user utilises a device which has been authenticated to the online server;

the user previously registered to the server whereby the server has a user record for the user comprising user ID; user public key; encrypted user private key;
the user indicates that they wish to register a card, supplying the card ID from the card;
the device having the user ID, user password, user public key and user private key;
the device then transmits the card ID to the online server in order to receive the card public key for that card identified by that card ID from the server:
the device then encrypts the user private key against the card public key to generate a card encrypted user private key;
the device uploads the card encrypted user private key to the online server where it is referenced against a user record on the online server;
thereby to enable a (third) user password reset step before the user;
said reset step comprising
the user loads to the device the recovery card password and (associated with) the card ID;
the device receives from the online server the encrypted user private key and the encrypted card private key
the device decrypts the encrypted card private key by using the recovery card password provided to the device by the user so as to obtain a recovered card private key;
the device decrypts the card encrypted user private key by using the recovered card private key in order to thereby obtain a recovered user private key;
the device encrypts the recovered user private key against a new user password provided to the device by the user as part of the password reset step;
thereby to produce a new encrypted user private key;
the device uploads the new encrypted user private key to the online server whereby the new encrypted private key replaces the encrypted private key previously associated with their user record corresponding to their user ID.

[0015] In a further broad form of the invention there is provided in a client server system a method of password replacement for a user using a device to communicate with the server;

the user authenticated with the server using a user set of credentials including a user ID and user password and user private key and user public key;

the device authenticated with the server using a device a set of credentials including a device ID and device password and device private key and device public key;
the method reliant on creation of a recovery card having its own set of credentials including a recovery card ID, a recovery card password, a recovery card private key and a recovery card public key;
the method comprising associating a selected recovery card with the user on the server whereby subsequently the user may utilise the device and selected ones of the recovery card credentials to replace selected ones of the user credentials on the server.

[0016] In yet a further broad form of the invention there is provided a method of secure access control and transmission of digital items in a client server environment wherein at least one server communicates with a plurality of participating user entities by means of participating digital devices in the form of digital communications devices in order to provide secure access control and transmission services for digital items created by, originated by or brought into the environment by ones of the plurality of participating user entities;
said method comprising:
for each participating user entity participating in the environment;
authenticating the each participating user entity as between the entity and the at least one server for that user entity in a participating user entity authentication step;
authenticating each participating digital device used by the participating user entity as between the digital device and the at least one server for that user entity in a participating digital device authentication step;
and wherein each participating user entity has a participating user entity encryption key;
said method comprising:
a sending (source) user entity enabling secure access for a receiving (target) user entity of the plurality of user entities to a designated (unencrypted) digital item which is recorded as under the custody of the sending (source) user entity;
said sending (source) user entity encrypting the designated digital item or the user entity encrypting a key by which the designated digital item has been encrypted so as to produce an

encrypted designated digital item or an encrypted digital item key preparatory to the sending user entity enabling secure access to the designated digital item by the receiving (target) user;
said sending (source) user entity utilising a sending (source) user entity digital device which has been authenticated as between the server and the sending (source) user entity thereby to enable the designated receiving (target) user entity to decrypt the designated digital item;
said sending user entity having a sending user entity encryption key;
said designated receiving user entity having a designated receiving user entity encryption key;
access to the designated digital item effected by the sending user entity encrypting the designated digital item so as to produce the encrypted designated digital item by utilising the designated receiving user entity encryption key to encrypt the designated digital item or access to the designated digital item effected by the sending user entity encrypting the document key of the designated digital item so as to produce the encrypted document key of the designated digital item by utilising the designated receiving user entity encryption key to encrypt the document key of the designated digital item.

[0017] Preferably said sending user entity receives said designated receiving user entity encryption key from said at least one server.

[0018] Preferably the steps of encryption or decryption are executed on the participating digital devices.

[0019] Preferably each digital device is authenticated by the system before it can be used within the system.

[0020] Preferably parameters associated with each user entity are stored on the server preparatory to each user entity participating in the system.

[0021] Preferably parameters associated with each digital item are stored on the server preparatory to each user entity participating in the system.

[0022] Preferably the digital item is stored on the server in encrypted form.

[0023] Preferably the digital item is stored on the server in encrypted form by reference to a document key.

[0024] In yet a further broad form of the invention there is provided a method of secure access control and transmission of digital items in a client server environment incorporating a first underlying methodology wherein servers forming part of the system never have enough information to decrypt the user entity content.

[0025] Preferably there is incorporated a second underlying methodology wherein all user entity content to which a user may have access is encrypted (directly or transitively) against their private key.

[0026] Preferably the servers forming the part of the system never see this key in an unencrypted form.

[0027] Preferably the servers store the key encrypted using a key derived from the user's password.

[0028] Preferably the servers forming part of the system never see the user's password (or any keys derived from it).

[0029] Preferably the user's password is never sent to the servers, even in encrypted form.

[0030] Preferably the user's password is verified at the client device level by using the derivative of the user's password to decrypt the user key.

[0031] Preferably systems implementing the method rely on an infrastructure which includes cryptographic keys.

[0032] Preferably cryptographic keys are associated with the various elements making up the systems.

[0033] Preferably cryptographic keys are associated with:
Each entity or user participating in the system;
Each digital device used to participate in the system;
Each digital item stored or transmitted within the system.

[0034] Preferably the cryptographic keys are utilised to encrypt and decrypt each digital item (typically a document) stored or transmitted within the system, and to determine which users may decrypt a digital item.

[0035] Preferably the method is further applied to a method for authentication of a user login request in order to permit an authenticated user session between the user and a server on a device of the user in a client/server system where the device and user are previously jointly authenticated with respect to the server;
the user having at least for a given user login: a user ID, a user password, a user private key and a user public key;
the device having at least a device ID, a device public key and a device private key;
the server in a user enrolment event for the user storing the public key of the user and a derivative of the user private key of the user against the user ID;
the server in a device enrolment event for the device storing the device public key against the device ID;
said method comprising authenticating login of the user in a login event by the user to the server by the following steps:
provided the server has enrolled the user login;
provided the server has enrolled the device;
the device signals to the server commencement of the login event by the user;
the server supplies to the device the derivative of the user private key stored against the user ID and a randomly created authentication challenge;
the device performs a comparison operation to determine if the password of the user is the same as the password used to create the derivative to encrypt the user private key, and thereby decrypts the user private key from the derivative;
if the comparison operation is positive the device signals to the server that the user login request has been authenticated by providing a cryptographic response to the authentication challenge which requires possession of the user private key, whereupon the server treats the login event as successful such that the user ID is now authenticated for the user session on the server using the device.

[0036] Preferably the comparison operation compares the user password supplied during the login attempt with the user password used to encrypt the user private key to generate the value stored on the server.

[0037] Preferably the derivative of the user private key is an encryption of the private key encrypted against a value derived from the user password.

[0038] In yet a further broad form of the invention there is provided a method of secure access control and transmission of digital items in a client server environment as claimed in any previous claim further incorporating a password recovery method utilising a recovery card, said password recovery method comprising
in a first card generation step
executed on an off-line device;
for each card the off-line device generates a card private key and
a card password and
a card ID and
a card public key (derived from the card private key);
the off-line device then encrypts the card private key using the card password so as to generate an encrypted card private key;
the off-line device then embosses and/or prints on the card so as to be readable by a user the card ID and the card password;
the system administrator may then upload to an online server (platform)
the card ID and
the encrypted card private key and
the card public key;
in a second user setup step the user utilises a device which has been authenticated to the online server;
the user previously registered to the server whereby the server has a user record for the user comprising user ID; user public key; encrypted user private key;
the user indicates that they wish to register a card, supplying the card ID from the card;
the device having the user ID, user password, user public key and user private key;
the device then transmits the card ID to the online server in order to receive the card public key for that card identified by that card ID from the server:
the device then encrypts the user private key against the card public key to generate a card encrypted user private key;

the device uploads the card encrypted user private key to the online server where it is referenced against a user record on the online server;
thereby to enable a (third) user password reset step before the user;
said reset step comprising
the user loads to the device the recovery card password and (associated with) the card ID;
the device receives from the online server the encrypted user private key and the encrypted card private key
the device decrypts the encrypted card private key by using the recovery card password provided to the device by the user so as to obtain a recovered card private key;
the device decrypts the card encrypted user private key by using the recovered card private key in order to thereby obtain a recovered user private key;
the device encrypts the recovered user private key against a new user password provided to the device by the user as part of the password reset step;
thereby to produce a new encrypted user private key;
the device uploads the new encrypted user private key to the online server whereby the new encrypted private key replaces the encrypted private key previously associated with their user record corresponding to their user ID.

[0039] In yet a further broad form of the invention there is provided a method of secure access control and transmission of digital items in a client server environment;
the method further including a method of password replacement for a user using a device to communicate with the server;
the user authenticated with the server using a user set of credentials including a user ID and user password and user private key and user public key;
the device authenticated with the server using a device a set of credentials including a device ID and device password and device private key and device public key;
the method reliant on creation of a recovery card having its own set of credentials including a recovery card ID, a recovery card password, a recovery card private key and a recovery card public key;
the method comprising associating a selected recovery card with the user on the server whereby subsequently the user may utilise the device and selected ones of the recovery card

credentials to replace selected ones of the user credentials on the server.

[0040] In yet a further broad form of the invention there is provided a client server environment operating according to the method as described above.

[0041] In yet a further broad form of the invention there is provided method for authentication of a user login request in order to permit an authenticated user session between the user and a server on a device of the user in a client/server system where the device and user are previously jointly authenticated with respect to the server;

the user having at least for a given user login: a user ID, a user password, a user private key and a user public key;

the device having at least a device ID, a device public key and a device private key;

the server in a user enrolment event for the user storing the public key of the user and a derivative of the user private key of the user against the user ID;

the server in a device enrolment event for the device storing the device public key against the device ID;

said method comprising authenticating login of the user in a login event by the user to the server by the following steps:

provided the server has enrolled the user login;

provided the server has enrolled the device;

the device signals to the server commencement of the login event by the user;

the server supplies to the device the derivative of the user private key stored against the user ID and a randomly created authentication challenge;

the device performs a comparison operation to determine if the password of the user is the same as the password used to create the derivative to encrypt the user private key, and thereby decrypts the user private key from the derivative;

if the comparison operation is positive the device signals to the server that the user login request has been authenticated by providing a cryptographic response to the authentication challenge which requires possession of the user private key, whereupon the server treats the login event as successful such that the user ID is now authenticated for the user session on the server using the device.

[0042] Preferably, the comparison operation compares the user password supplied during the login attempt with the user

password used to encrypt the user private key to generate the value stored on the server.

[0043] Preferably, the derivative of the user private key is an encryption of the private key encrypted against a value derived from the user password.

[0044] In yet a further broad form of the invention there is provided a client server environment operating according to the method of any previous claim.

[0045] In yet a further broad form of the invention there is provided a device operating according to the method as described above.

[0046] In yet a further broad form of the invention there is provided a digital item secured according to the method as described above.

[0047] In yet a further broad form of the invention there is provided media containing code which when executed by a processor performs the method as described above.

[0048] In yet a further broad form of the invention there is provided a recovery card marked with indicia and operable according to the method as described above.

BRIEF DESCRIPTION OF DRAWINGS

[0049] Embodiments of the present invention will now be described with reference to the accompanying drawings wherein:

[0050] Figure 1A is a block schematic diagram of an initial authentication procedure for authenticating a user entity in order that the client entity may participate in the system of the present invention.

[0051] Figure 1B is a block schematic diagram of an initial authentication procedure for authenticating a client device in order that the client device may participate in the system of the present invention.

[0052] Figure 2 is a logic flowchart of a preferred method of authenticating the client device of figure 1B.

[0053] Figure 3A is a block schematic diagram of a first part of a recovery card procedure and system according to an embodiment of the present invention

[0054] Figure 3B is a block schematic diagram of a second part of a recovery card procedure and system according to an embodiment of the present invention,

[0055] Figure 4 is a block schematic diagram of a third part of a recovery card procedure and system according to an embodiment of the present invention,

[0056] Figure 5A is a flowchart of a recovery card generation procedure according to an embodiment of the present invention,

[0057] Figure 5B is a flowchart of the setup procedure for associating a card with a user record in accordance with an embodiment of the present invention,

[0058] Figure 5C is a flowchart of a reset procedure for re-encrypting a user key against a new password in accordance with an embodiment of the present invention.

[0059] Figure 6 is a block schematic diagram of a user entity password authentication according to an embodiment of the present invention and without server knowledge of user entity password or derivative thereof.

[0060] Figure 7 is a block schematic diagram of a user password reset procedure according to an embodiment of the present invention.

[0061] Figure 8A is a logic flowchart of a set up procedure for the password reset procedure of figure 4.

[0062] Figure 8B is a logic flowchart of the password reset procedure of figure 7.

[0063] Figure 9A is a block schematic diagram of an embodiment of an access and security control infrastructure within which embodiments of the present invention operate.

[0064] Figure 9B is a block schematic diagram of an embodiment of the access and security control infrastructure within which embodiments of the present invention operate.

[0065] Figure 10 is a logic flowchart of a preferred method of operation of the access and security control infrastructure of figure 9B.

[0066] Figure 11 is a system architecture block diagram of an example system to which the password recovery system of figures 3, 4, 5 and the user authentication system of figures 6,7,8 and access and security control infrastructure of the figures 9 and 10 may be applied.

DESCRIPTION OF EMBODIMENTS

Security Principles

[0067] A first underlying methodology is that servers forming part of the system never have enough information to decrypt the user entity content.

[0068] A second underlying methodology is that all user entity content to which a user may have access is encrypted (directly or transitively) against their private key. The servers forming the part of the system never see this key in an unencrypted form. In particular forms the servers store the key encrypted using a key derived from the user's password.

[0069] A third underlying methodology is that the servers forming part of the system never see the user's password (or any keys derived from it). More generally the user's password is never sent to the servers, even in encrypted form. Again, more generally the user's password is verified at the client device level by using the derivative of the user's password to decrypt the user key.

Cryptographic Keys

[0070] Systems implementing aspects of embodiments of the present invention rely on an infrastructure which includes cryptographic keys. More specifically, cryptographic keys are associated with the various elements making up the systems. Keys are associated with:

Each entity or user participating in the system;

Each digital device used to participate in the system;

Each digital item stored or transmitted within the system.

[0071] These keys are utilised to encrypt and decrypt each digital item (typically a document) stored or transmitted within the system, and to determine which users may decrypt a digital item.

User and device authentication process

[0072] Figure 1A is a block schematic diagram of an initial authentication procedure for authenticating a user entity in order that the client entity may participate in the system of the present invention.

[0073] Figure 1B is a block schematic diagram of an initial authentication procedure for authenticating a client device in order that the client device may participate in the system of the present invention.

[0074] Figure 2 is a logic flowchart of a preferred method of authenticating the client device of figure 1B.

[0075] The above described infrastructure forms the basis for a document storage and access control system in accordance with embodiments of the invention as will be described immediately below.

Password recovery using pseudonymous cryptographic keys.

[0076] Figure 3A is a block schematic diagram of a first part of a password recovery card procedure and system according to an embodiment of the present invention

[0077] Figure 3B is a block schematic diagram of a second part of a recovery card procedure and system according to an embodiment of the present invention,

[0078] Figure 4 is a block schematic diagram of a third part of a recovery card procedure and system according to an embodiment of the present invention,

[0079] Figure 5A is a flowchart of a recovery card generation procedure according to an embodiment of the present invention,

[0080] Figure 5B is a flowchart of the setup procedure for associating a card with a user record in accordance with an embodiment of the present invention,

[0081] Figure 5C is a flowchart of a reset procedure for re-encrypting a user key against a new password in accordance with an embodiment of the present invention

[0082] Stated in summary form: This is the recovery card mechanism.

[0083] Password recovery is complex because you want to make it convenient for users to regain access if they forget their password, but without reducing the overall security of the scheme.

[0084] At the same time, it's important that we can't reset passwords ourselves, as that would undo our zero-knowledge promise, as if we can reset a customer password, we can then grant ourselves access.

[0085] We have therefore devised a password recovery system using pseudonymous cryptographic keys to ensure that a user can reset their password, but that our servers cannot.

[0086] In order to recover the user's password three pieces of information are required:
(1) The recovery card password, printed on the card; and
(2) The recovery card private key, encrypted against the recovery card password; and
(3) The user's private key, encrypted against the recovery card private key.

[0087] During normal operation, no one entity has all three pieces of information:

The recovery card generation system has the card password and card private key, but not the user's private key. (Note that this system must be permanently isolated from any network in order to protect the security of the scheme.)

The Lockbox platform has the card private key (encrypted, per 2) and the user's private key (encrypted, per 3), but not the recovery card password.

The user has the recovery card password, but not the other two pieces of information.

[0088] Thus, if any one of those entities is compromised alone (e.g. if the recovery card is lost), the attacker cannot reset the customer password. Instead, a trusted device must be used to recover items (2) and (3) from the Lockbox Platform, and then the user provides (1). This then provides access to the user's private key (using 1 to decrypt 2, and 2 to decrypt 3), allowing the device to set a new password by re-encrypting and uploading the user's private key to the Platform.

[0089] We have further enhanced the security of the scheme by making the recovery cards pseudonymous rather than customised for a user; bulk distribution of generated cards to a customer for then distribution to their users makes it impractical for Lockbox staff to manipulate which recovery card is associated with a given user if the secrecy of the recovery card generation system was compromised.

[0090] The above described security card process and related password recovery system of embodiments of the present invention in relation to figures 3, 4, 5 may be utilised to advantage in combination with the user/password authentication system and the digital item access control systems referred to and described below:

User entity password authentication without server knowledge of password or derivative but allowing password reset

[0091] Figure 6 is a block schematic diagram of a user entity password authentication according to an embodiment of the present invention and without server knowledge of user entity password or derivative thereof.

[0092] Figure 7 is a block schematic diagram of a user password reset procedure according to an embodiment of the present invention.

[0093] Figure 8A is a logic flowchart of a set up procedure for the password reset procedure of figure 7.

[0094] Figure 8B is a logic flowchart of the password reset procedure of figure 7.

[0095] The methodology reflected in figures 6,7,8A,8B may be summarised as follows:

[0096] Legacy password authentication schemes work by storing the user's password in the server. When the user attempts to login, the server compares the supplied password to the stored one, and permit login on a match.

[0097] Modern password authentication schemes do not store the user's password, but instead store something derived from the password that can't easily be turned back into the password (e.g. a cryptographic hash of the password). The server then uses the same operation to compute the derivative of the supplied password and compares the output to the stored value. While this is very strong, it still involves the storage of something derived from the password on the server which represents a vulnerability.

[0098] Our scheme does not store passwords or derivatives on the server. Instead, we use cryptographic keys to authenticate the user. When the user account is created, the user's device creates a public-private cryptographic key. A copy of the private key is encrypted against a value derived from the user's password (a hash of it computed using the PBKDF#2 scheme), and uploaded to our server. The device also generates and stores a public and private key, stores the private key on itself and sends the public key to the server.

[0099] When the user logs into their account, the server requires either a device public key or a strong authentication event, before sending the encrypted user private key to the device. The user then supplies their password to the device and the device derives the value and uses that to decrypt the key. If that operation succeeds, the user has proven they know the correct password; the device then proves possession of the private key to the server, completing the login event.

[00100] We support password reset by allowing the (target) user to specify which other (source) users may reset their password. We then store a copy of the target user's private key encrypted against the source user's public key. This allows the source user to decrypt the user's private key and re-encrypt it against a new password, allowing the target user to login with that new password. Note that the trusted device

or strong authentication requirement means that the source user cannot simply login as the target user unless they also have access to a trusted device and/or the target user's strong authentication mechanism.

[00101] The above described process with reference to figures 6,7,8 in relation to user authentication may be utilised to advantage in combination with a digital item access control system as to be described below.

Digital item access control system

[00102] The above described infrastructure forms the basis for the access control and transmission system in accordance with embodiments of the present invention. The system seeks to control access to digital items circulated within the system. The system also seeks to secure the items during transmission and storage. An item will typically be a document. However, the item may be any block of digital data in respect of which it is desired to control access and also to secure the item during transmission to provide the access and also during storage prior to and after transmission.

[00103] The concepts which underlie the access and control system include:

1. Zero knowledge storage concept:

[00104] The platform which orchestrates the storage and transmission of digital items at no stage has an unencrypted form of the digital item stored on it.

[00105] More particularly in preferred forms the platform never receives or stores private keys or keys used to encrypt items. In preferred forms the platform may store public keys - but in this instance only to facilitate encryption elsewhere in the system.

[00106] Also, in preferred forms encryption of digital items or keys is never done on the platform.

2. The 'push' concept in access controls:

[00107] The owner entity of a document or a digital item (or user entity with access rights to the document or digital item) has to 'push' (which is to say grant access or instigate

transmission) to any other target entity or user which is seeking access to that document or digital item.

[00108] In preferred forms access involves receiving a key of the target entity or user which is then used by the owner entity to encrypt the document or digital item following which the thus encrypted document or digital item is stored within the platform and possibly transmitted to a device under the control of the target entity or user.

[00109] Transmission of the encrypted document or digital item may be by way of the platform.

[00110] In preferred forms the platform is implemented on a server. More preferably the server is a web-enabled server.

[00111] In no instance does the platform act as a target entity or user entity.

[00112] That is to say, the platform (most usually implemented as a server) is not relied on to enforce the access controls to digital items. In preferred forms cryptography enforces the access controls.

[00113] In preferred forms the steps of encryption and decryption in respect of cryptography are executed on digital devices.

[00114] In preferred forms each digital device is authenticated by the system before it can be used within the system.

[00115] Figure 9A is a block schematic diagram of a first embodiment of the access and security control infrastructure within which embodiments of the present invention operate.

[00116] With reference to figure 9A there is disclosed a first preferred embodiment of a secure access control and transmission system 10.

[00117] More particularly, the system 10 comprises a digital environment in this instance in the form of a client server environment made up of at least one server 11 and at least a

first client digital device 12 and a second client digital device 13.

[00118] The server 11 includes a memory 14 in communication with a processor 15 and in communication with a communication module 16.

[00119] First client digital device 12 includes a memory 17 in communication with a processor 18 and in communication with a communication module 19.

[00120] Second client digital device 13 includes a memory 20 in communication with a processor 21 and in communication with a communication module 22.

[00121] In preferred forms communication of data within this digital environment is by way of a packet switched network wherein each data packet 23 comprises an address portion 24 and a data portion 25.

[00122] In use if a second user in the form of target user 26 is to receive access to a first digital document 27 (the first digital document being under the control of a first user in the form of a source user 28) then a key [in this instance U2-K_{pub}]associated with the target user 26 is caused by the second client digital device 13 (in its capacity as the target user client device) to be transmitted in this instance from server 11 to first client digital device 12 (in its capacity as the source user client device). This key is used to encrypt the first digital document 27 following which the encrypted first digital document 27A is caused to be made available to the target user client device 13 in this instance via the digital environment including server 11.

[00123] Target user 26 may then cause the target user client device 13 to decrypt the encrypted first digital document 27A in order to obtain the first digital document 27 [in this instance using target user 2 private key U2-K_{PVT} under the RSA encryption system].

[00124] Figure 9B is a block schematic diagram of a second embodiment of the access and security control infrastructure within which embodiments of the present invention operate.

[00125] In this instance access to the document is controlled by controlling access to a document key by which the document is encrypted.

[00126] Figure 10 is a logic flowchart of a preferred method of operation of the access and security control infrastructure of figure 9B.

[00127] In one form and more particularly with reference to figure 10 the steps in the above described process of figure 9B are:

[00128] The source user 28 decides to provide access to the first digital document 27 to the target user 26.

[00129] Source user 28 selects the first digital document 27 and clicks 'grant access' on the source user client device 12.

[00130] Source user client device 12 retrieves file metadata of the first digital document 27 and a key [in this instance public key U2-K_{pub}] associated with the target user 26 from the server 11.

[00131] Source user client device 12 decrypts the document key of the first digital document 27 using a private key [in this instance U1-K_{pVT}] associated with the source user 28.

[00132] Source user client device 12 then encrypts the document key of the first digital document 27 using the key U2-K_{pub} associated with the target user 26 to generate a re-encrypted document key for first digital document 27.

[00133] Source user client device 12 adds the user ID of the target user and the re-encrypted document key for the first digital document 27 into a data packet which is transmitted to the server and saved referenced against the first digital document 27 within the digital item (file) list.

[00134] The target user entity 26 may then cause the target user client device 13 to decrypt first digital document 27 by unencrypting the re-encrypted document key by use of the target user's private key U2-K_{PVT}.

[00135] By thereby having been granted access to first digital document 27 of source user 28 and as now reflected in the metadata stored on the server 11 the target user now has the capability to make the first digital document 27 available to other users within the environment utilising the same procedure.

Example 1

[00136] Figure 11 is a system architecture block diagram of an example system to which the password recovery system of figure is 3, 4, 5 and user authentication of figures 6, 7, 8 and access and security control infrastructure of the figures 9 and 10 may be applied.

[00137] Cryptographic enforcement of access controls with zero knowledge storage.

[00138] Cryptographic enforcement of access controls means that each user has an encryption key, and documents are (transitively) encrypted against them. Users grant access to a document for other users by encrypting it against the other user's key, making it possible for the target user to decrypt it. That means that if the server isn't relied on to enforce the access controls to documents, as the cryptography enforces it instead.

[00139] Zero knowledge storage means that our servers store documents and supply them to users but cannot see inside the documents because they are encrypted. This is because the documents are encrypted (above) and our server is not granted access.

[00140] In preferred forms the keys are generated as summarised in the below table:

Name	Type	Use	Generated	Stored
platform	RSA	protect against data loss	HSM	HSM
user	RSA	allow users to access items	client side on first login	encrypted in platform
user-sensitive	AES	allow items to be encrypted such that they're not accessible to an impersonating user	client side on first login	encrypted in platform
device	RSA	allow the platform to authenticate a device, locally cache items	client side on first login	device
item	AES	protect an item before upload	client side on upload	with the item, encrypted against user keys

[00141] Device public and private keys (Kdevice) are generated and stored on the device.

[00142] A user's public key, but not the private key (Kuser), is stored on the device.

[00143] When a device first authenticates it downloads and caches the user private key (in encrypted form) from the platform. This key is kept in RAM and never stored on the device.

[00144] For each user, the platform stores:
RSA(Kplatform, (AES(PBKDF#2(users-password), Kuser))
RSA(Kplatform, (AES(PBKDF#2(users-password), Kuser-sensitive)).

[00145] When the platform has authenticated a device and wants to send it the user's key, note that it returns:
RSA(Kdevice, (AES(PBKDF#2(users-password), Kuser))

New device for an existing user (e.g. adding a second iPad or replacement device)

[00146] If an existing user attempts to log into the platform (termed 'Athena' in figure 5) on a new device (i.e. We already know they have been logging in on other device(s)) the authentication is halted and the user is notified that the administrators must approve the device, unless;

a) If the user has registered on another device of a specific type, then we send a push notification to that device with a generated authentication code or;
b) If a mobile phone number is on file for the user, a secondary authentication code is generated and sent via SMS for
validation of the new device else;
c) If no phone number is available, all of the admins associated with the user are notified and any of them can approve the login. Once approved a push notification is sent to the device advising the user they can login.

Keys on the Device

[00147] Kdevice public and private keys are generated and stored on the device.

[00148] Kuser's public key, but not the private key, is stored on the device.

[00149] When a device first authenticates it downloads and caches Kuser (in encrypted form) from the platform. This key is kept in RAM and never stored on the device.

New user, new device

[00150] A user within an Athena organisation that has the 'user admin' permission creates a new user account, specifying username. They will also be required to enter the user's email and mobile number.

[00151] The Athena platform generates an initial password for the user, format will be "word-word-word-word" where word is a three to six-character common word in English, e.g. "seeks-world-hop-strung".

[00152] The Athena platform creates an entry in the newUsers table, with the AthenaOrganisationID, username, PBKDF#2(password) for the user, as well as the contact details if provided. The platform returns [AthenaOrganisationID, username, initial-password] to the portal, which returns them to the admin user.

[00153] The admin user provides these three attributes to the end user, along with information on how to download the Athena app. The portal will create a generated PDF that has

the end user's username, initial password, AthenaOrganisationID and detailed app installation and download instructions. The content of the email will also include links to the appropriate app store download location. The email is, by default, emailed directly to the new user. The email can be disabled by the admin user. If the admin user elects to disable the welcome email, the document is available for download or resend at any time.

[00154] The end user downloads the Athena app. On first run, it prompts the user for an Athena organisation ID, their username and their initial password. If it's a new device, it hasn't yet generated or registered a device ID.

[00155] The device generates a 2048-bit RSA key Kdevice, a 2048-bit RSA key Kuser, and a 256-bit AES key Kuser-sensitive. It also generates a UUID, device ID.

[00156] The device sends a request to the Athena platform via / Register, containing device ID.

[00157] The platform confirms that the proposed device ID is unique. If so, it returns the Kplatform public key to the device. The device then prompts the user to set a new password, according to the current password complexity policy. It generates a new 256 bit AES key called KUserTransport, and then calculates
a. AES(KUserTransport, (AES(PBKDF#2(users-password), Kuser))
b. AES(KUserTransport, (AES(PBKDF#2(users-password), Kusersensitive))
c. RSA(KPlatform, KUserTransport) 9. The device then sends a user registration request to the platform, signed by the device key, containing:
a. deviceID;
b. (AthenaOrganisationID, username, RSA(Kplatform, initial-password))
c. RSASign(Kdevice, (AES(KUserTransport, (AES(PBKDF#2(new-password), Kuser)), AES(KUserTransport, (AES(PBKDF#2(new-password), Kusersensitive)), RSA(KPlatform, UserTransport)).

[00158] The platform decrypts that blob, calculates PBKDF#2(initial-password), and verifies whether there is a row in newUsers that matches those details. (If not, return error message and fail out.) If so, it creates new records in the

AthenaUsers table to reflect the user account, the AthenaDevices table to reflect the device, and the AthenaUsersDevices table to reflect that that device is associated with that user. It then returns a success message to the device.

[00159] The platform verifies the device key signature, and then stores the two encrypted passwords (still encrypted against Kplatform) and the encrypted KUserTransport in the user table.

[00160] The platform notifies the user by email and/or SMS that a new device has been registered to their Athena account, and to report immediately to their organisation's administrator if they don't recognise the activity. Depending on the Athena organisation's preferences, a notification to the user admin may also be required. All new device additions are logged and are reported to administrators at next login according to the alert preference set for the admin account.

New user on an existing device (e.g. adding a second board)

[00161] Essentially, the flow above, but initiated from the '+' tile on the Athena home page.

[00162] Flow therefore after step 5 (user admin sends password to user) jumps immediately to step 10 in section 3.7 ("The device then sends deviceID and (AthenaOrganisationID, username, RSA(Kplatform, password))").

[00163] For the web interface, a browser is a device, and subject to device registration on login. That means 2FA at login, and it then generates a Kdevice to use to receive and access the data. For the sake of user experience, we can save Kdevice in the local browser (local storage, rather than a cookie, preferably) so that the same user using the same browser doesn't need to go through the registration flow more than once in a blue moon. Each time the session times out, they'll need to re-enter their password. Not sure yet how often 2FA might be required, but less often than that.

[00164] Those Kdevice registrations for web browsers should probably timeout in the backend (30 days after last used?), to avoid the table blowing out if someone uses lots of different browsers. (That "expiring, temporary shadow user" feature we

were talking about for international travel would also be useful for users' needing emergency access from an untrusted device, e.g. in a hotel business centre.)

New device for an existing user (e.g. adding a second iPad or replacement device)

[00165] If an existing user attempts to log into Athena on a new device (i.e. We already know they have been logging in on other device(s)) the authentication is halted and the user is notified that the administrators must approve the device, unless;

a) If the user has registered on another device (iPhone or Windows 10), then we send a push notification to that device with a generated authentication code or;

b) If a mobile phone number is on file for the user (we will strongly recommend that it is), a secondary authentication code is generated and sent via SMS for validation of the new device else;

c) If no phone number is available, all of the admins associated with the user are notified and any of them can approve the login, once approved a push notification is sent to the device advising the user they can login.

[00166] This secondary validation of new devices is the default but can be disabled by admin. We will maintain a list of device hashes for each approved device.

Normal Authentication Flow - online

[00167] When a user on a device logs into Athena from a device that has a Kdevice:

1. The device submits its Athena UUID and the user's UUID to the login API.
2. The platform verifies that that device is associated with that user and, if so, generates a 256-bit random number called a nonce, and sends RSA(Kdevice, nonce) to the device.
3. The device decrypts the nonce, and returns SA(Kplatform, nonce).
4. If the platform decrypts the same nonce as it sent to that device, the platform retrieves the encrypted user key from its data store and its associated AES key KUserTransport which has been encrypted by KPlatform, unwraps the outer RSA encryption on KUserTransport using Kplatform and re-encrypts it against KDevice. It then sends:

a. AES(KUserTransport, AES(PBKDF#2(user-password), Kuser))
b. AES(KUserTransport, AES(PBKDF#2(user-password), Kuser-sensitive
c. RSA(KDevice, KUserTransport)

5. The device decrypts KUserTransport using KDevice, then decrypts the outer AES layer using this decrypted key on the other properties.
6. The device prompts the user for their password, calculates PBKDF#2(userpassword), and then uses this to decrypt Kuser and Kuser-sensitive.
7. The device uses the recovered Kuser private key to create RSA(Kplatform, RSASign(Kuser, nonce)), and sends it to the platform.
8. The platform verifies the signature on the nonce using the user's public key, verifies that the nonce was the one sent to the device, and verifies that that device is associated with that user's account. If all those checks pass, the server creates a random session token (1024-bit random number?), creates an entry in the session table with that session token , user UUID, device UUID, and other appropriate metadata (e.g. session start timestamp, session expiry timestamp). It then returns session token to the device.
9. All requests from that device to any authenticated API endpoint include the session token. When each request is received, the platform checks that the session token is in the session table and has not expired; if so, it retrieves the session context. The platform also validates that the session token has access to the specified API, which may not be true even if the session token is valid.

[00168] We don't need to prompt for a username on login. The only times we need it is when we are adding a new device to a user or adding a new board/company association for an existing user. For the next version, I might suggest moving the offline verification code generation process 3.10 section. That way, every time we verify the password online, we create a new copy of the offline verification code, which should reduce the chance for a password mismatch around changes, but it potentially adds some complexity/time to the online login process, so we might want to throttle it to just once a day.

Normal Authentication Flow - offline

[00169] If the user attempts to use Athena when there is no internet connection the following is the flow. This will only work if they have authenticated at least once (they won't have any data if they haven't anyway).

[00170] A nonce is generated on the device and used to securely store (keychain or equivalent) RSA(KDevice, nonce), AES(PBKDF#2(user-password), nonce). This is stored only once and is created during the first successful online authentication.

[00171] The nonce should only be used for offline authentication (and nothing else) and must be regenerated every time the password changes.

Unsuccessful Authentication Flow

[00172] If there is an unsuccessful login attempt the Athena app will allow two subsequent attempts (i.e. 3 failed logins' in succession) and then delay subsequent login attempts by five minutes. If there is a second set of three failed logins', Athena will delete all app data (revert to an initial application state).

[00173] All of the above actions are immediately logged and reported back to the Athena server. Each customer (secretariat or administrator) will be independently notified of the event via their selected notification method (email, SMS or PNS).

User Information Recorded

[00174] For each user added to the system by a board administrator we will require the following to be captured:

- First and Last name
- Email address
- Mobile phone number
- And, if a company employee a unique employee number.

User Password Reset

[00175] Should a user forget their password there will be a password reset mechanism.

[00176] On the login screen for the iOS and Win 10 apps to initiate password reset. This reset function will require the use of a plastic identity card with a serial number and an

authentication code printed on the rear. This card will be sent by the customer (secretariat) to each Athena user on initial registration by the administrators (along with some spares). The codes are for one use only. The customer, on receipt of their card(s) and while still remembering their password, registers the card(s) with their account through the app (details in 0 below).

[00177] The process will be that the user taps on the reset password button and they are prompted to either scan the QRCode (via the camera on the device) or enter the authentication code.

[00178] Once the authentication code has been entered, immediate entry of a new password (of appropriate strength) will then be required. All admins will be alerted
[00179] of the password change according to the alert preferences set.

[00180] If a user has lost their card any of the admins associated with the can request a new card be sent to the user.

[00181] All other admins associated with the user will be alerted of the password change according to the alert preferences set for that account.

[00182] When a card with an authentication code is used a notification is provided to Lockbox administration staff to assign and send a new card to the user.

Card generation process:
[00183] The system generates tuples of (card serial number, card password, card private RSA key, card public RSA key). It exports card printing records of the form [(card serial number, card password)].

[00184] It exports card upload records of the form [(card serial number, KCard-public, PKCS#8(card-password, KCard))], which are uploaded to the platform. The card private keys are protected using a PKCS#8 format, with the private key encrypted against the card-password using aes128CBC scheme. Both the public key and private key are exported in PEM format.

[00185] The platform then stores: card-serial-number: (KCard-public, PKCS#8(cardpassword, Kcard)) <-- that is, if someone has the card serial and password, they can access the private key. Else nobody can, not even the platform.

[00186] Card registration process: When a user receives a card, they can add it to their account. This is part of the initial registration flow for the user. When a user adds a card, it prompts just for the card serial number, entered twice for certainty, or via QR code. The app then asks the Athena platform for the card-public-key (Kcard-public) associated with that card serial number. It also calculates new AES key called Ktransport and KCardWrap. The app then uses this to calculate: card-serial-number: (RSA(KPlatform, Ktransport), RSA(KCard, KCardWrap), AES(Ktransport, AES(KCardWrap, (Kuser, Kuser-sensitive))) It then sends that to the platform.

[00187] The platform will then unwrap Ktransport, unwrap the data encrypted by these AES keys.

[00188] The platform then:

(i) confirms that that card is not issued to any other user.

(ii) updates the card record to reflect that the card has been issued to that user.

(iii) updates the user record in Lockbox to reflect the card's access by appending the data above.

The resulting Lockbox user record looks like: user-id: (RSA(KPlatform, KUserTransport), AES(KUserTransport(AES(PBKDF2(password), Kuser))), AES(KUserTransport(AES(PBKDF2(password), Kuser-sensitive))), [impersonator-user-id: RSA(KUser-impersonator, KUserTransport-impersonator), AES(KUserTransport, AES(KUserTransport-impersonator, Kuser))], #users allowed to impersonate/reset [card-serial-number: RSA(KCard, KCardWrap), AES(KUserTransport, AES(KCardWrap, (Kuser, Kuser-sensitive)))]) and the Lockbox card record looks like: card-serial-number: (KCard-public, PKCS#8(card-password, KCard), [user-id]

[00189] Note that if the person is an Athena user, the Athena platform will also store: person-id: ([board-id: user-

id],[card-serial-number]) This allows the Athena platform to automatically register a new board membership (and therefore a new Kuser) with the same card for password recovery.

[00190] Password recovery process with a card:
1. User chooses "Lost password".
2. User selects "I have my registered card".
3. App prompts for the card serial number and password (or QR code).
4. App requests the encrypted card private key from the platform, supplying both its user-id and device-id.
5. Platform verifies that the both the user and the device are known, current and associated with each other. It then retrieves three elements from the user record: RSA(KPlatform, KUserTransport), RSA(KCard, KCardWrap), AES(KUserTransport, AES(KCardWrap, (Kuser, Kuser-sensitive)))
6. From the card record, the platform retrieves: PKCS#8(card-password, KCard)
7. The platform then generates a new 256 bit AES key we will call Ktransport. It uses KPlatform to decrypt KUserTransport, and then uses KUserTransport to decrypt the last element above before re-encrypting it against Ktransport. It then returns to the device: RSA(KDevice, Ktransport), AES(Ktransport, PKCS#8(card-password, KCard)), RSA(KCard, KCardWrap), AES(KCardWrap, (Kuser, Kuser-sensitive)))),
8. The app uses Kdevice to decrypt the payload, and then the card password to decrypt KCard, and then the card private key to decrypt Kuser.
9. The app prompts the user for a new password.
10. The app calculates a new 256 bit AES key KUserTransport and then calculates RSASign(KCard, (user-id, card-serial-number, AES(KUserTransport(PBKDF2(password, Kuser))), AES(KUserTransport(PBKDF2(password, Kuser-sensitive)))), RSA(KPlatform, KUserTransport)), and sends that back to the platform, to update the user record.
11. The platform will verify that the card is associated with that user, verify the signature using KCard-Public, and update the user record.
12. The platform confirms the user record was updated. [For Athena, if the user has multiple memberships and therefore multiple Kuser records, the app repeats steps 4-12 for each of those Kuser's with the same card key.]

13. The app tells the platform that the card has been used to recover everything required, and that the card should be invalidated.
14. The platform marks the card as used. Whilst the card has been marked as used it is still available for use for a 14-day period from that time. Once the 14 days has expired the card is marked as terminated and can no longer be used. When a card is used an automatic process is initiated to send a new card to the user.
15. The app notifies the user that they should immediately request a new card from one of their secretariats and register it in the app immediately once it arrives.

[00191] Password recovery process without a card:
User chooses "Lost password".
User selects "I do not have a card or I have lost my card".
Athena app prompts user to confirm that all annotations and any other "eyes only" content will be lost in the password reset; encourages user to look for their card and/or remember their password if they can. User confirms "I understand".
App sends a "password reset" request to Athena platform, which does a 2FA of the user to confirm it's not an attacker attempting to reset the user's password. Then confirms request to the app, with a temporary session token (or equivalent).
App then prompts user for a new password and calculates PBKDF#2(newpassword).
App then does: i. for each board that the user is a member of, retrieve the secretariat generic user's Kuser's public key (which we'll call, for this piece, Ksecretariat); ii. calculate RSA(Ksecretariat, PBKDF#2(new-password)) for each Ksecretariat; iii. sends the list of [(board-id), RSA(Ksecretariat, PBKDF#2(newpassword))] to the platform, authenticated with the session token .
The platform then notifies each secretariat that a password change is required for that user. As each secretariat approves the change, their app uses their Kuser to decrypt the Ksecretariat for the board and the Kuser for the target user. It uses these to decrypt the PBKDF#2(new-password), and reencrypt the Kuser for the target user to generate RSA(Kplatform, AES(PBKDF#2(new-password), Kuser)) for the target user, and then lodge this as the new platform copy of the Kuser for the target user.

[00192] As this happens for each board the user is a member of, their documents will become available again as they regain access to their Kuser records using their new password.

[00193] Broadly this means that if you have your password, you can register a new card. If you have a registered card, you can reset your password. If you have neither, you lose some super-sensitive data (annotations, notes and backups), and you have to have the secretariats you're associated with approve the reset. If you're only a member of one board, that's one reset approval. If you're a member of a dozen, then each approves it. That's necessary so that a secretariat at one board can't reset a user's credential and then see all of the other boards' content that that user has access to.

INDUSTRIAL APPLICABILITY

[00194] Embodiments of the present invention operate to provide controlled and secure access to and transmission of digital items (which may be in the form of documents) as between entities or users of a system.

CLAIMS

1. A password recovery method utilising a recovery card, said method comprising
in a first card generation step
executed on an off-line device;
for each card the off-line device generates a card private key and
a card password and
a card ID and
a card public key (derived from the card private key);
the off-line device then encrypts the card private key using the card password so as to generate an encrypted card private key;
the off-line device then embosses and/or prints on the card so as to be readable by a user the card ID and the card password;
the system administrator may then upload to an online server (platform)
the card ID and
the encrypted card private key and
the card public key;
in a second user setup step the user utilises a device which has been authenticated to the online server;
the user previously registered to the server whereby the server has a user record for the user comprising user ID; user public key; encrypted user private key;
the user indicates that they wish to register a card, supplying the card ID from the card;
the device having the user ID, user password, user public key and user private key;
the device then transmits the card ID to the online server in order to receive the card public key for that card identified by that card ID from the server:
the device then encrypts the user private key against the card public key to generate a card encrypted user private key;
the device uploads the card encrypted user private key to the online server where it is referenced against a user record on the online server;
thereby to enable a (third) user password reset step before the user;
said reset step comprising
the user loads to the device the recovery card password and (associated with) the card ID;

the device receives from the online server the encrypted user private key and the encrypted card private key
the device decrypts the encrypted card private key by using the recovery card password provided to the device by the user so as to obtain a recovered card private key;
the device decrypts the card encrypted user private key by using the recovered card private key in order to thereby obtain a recovered user private key;
the device encrypts the recovered user private key against a new user password provided to the device by the user as part of the password reset step;
thereby to produce a new encrypted user private key;
the device uploads the new encrypted user private key to the online server whereby the new encrypted private key replaces the encrypted private key previously associated with their user record corresponding to their user ID.

2. In a client server system a method of password replacement for a user using a device to communicate with the server;
the user authenticated with the server using a user set of credentials including a user ID and user password and user private key and user public key;
the device authenticated with the server using a device set of credentials including a device ID and device password and device private key and device public key;
the method reliant on creation of a recovery card having its own set of credentials including a recovery card ID, a recovery card password, a recovery card private key and a recovery card public key;
the method comprising associating a selected recovery card with the user on the server whereby subsequently the user may utilise the device and selected ones of the recovery card credentials to replace selected ones of the user credentials on the server.

3. A method of secure access control and transmission of digital items in a client server environment wherein at least one server communicates with a plurality of participating user entities by means of participating digital devices in the form of digital communications devices in order to provide secure access control and transmission services for digital items created by, originated by or brought into the environment by ones of the plurality of participating user entities;

said method comprising:
for each participating user entity participating in the environment;
authenticating the each participating user entity as between the entity and the at least one server for that user entity in a participating user entity authentication step;
authenticating each participating digital device used by the participating user entity as between the digital device and the at least one server for that user entity in a participating digital device authentication step;
and wherein each participating user entity has a participating user entity encryption key;
said method comprising:
a sending (source) user entity enabling secure access for a receiving (target) user entity of the plurality of user entities to a designated (unencrypted) digital item which is recorded as under the custody of the sending (source) user entity;
said sending (source) user entity encrypting the designated digital item or the user entity encrypting a key by which the designated digital item has been encrypted so as to produce an encrypted designated digital item or an encrypted digital item key preparatory to the sending user entity enabling secure access to the designated digital item by the receiving (target) user;
said sending (source) user entity utilising a sending (source) user entity digital device which has been authenticated as between the server and the sending (source) user entity thereby to enable the designated receiving (target) user entity to decrypt the designated digital item;
said sending user entity having a sending user entity encryption key;
said designated receiving user entity having a designated receiving user entity encryption key;
access to the designated digital item effected by the sending user entity encrypting the designated digital item so as to produce the encrypted designated digital item by utilising the designated receiving user entity encryption key to encrypt the designated digital item or access to the designated digital item effected by the sending user entity encrypting the document key of the designated digital item so as to produce the encrypted document key of the designated digital item by utilising the designated receiving user entity encryption key to encrypt the document key of the designated digital item.

4. The method of claim 3 wherein said sending user entity receives said designated receiving user entity encryption key from said at least one server.

5. The method of claim 3 or 4 wherein the steps of encryption or decryption are executed on the participating digital devices.

6. The method of claim 3 or 4 or 5 wherein each digital device is authenticated by the system before it can be used within the system.

7. The method of any one of previous claims 3 to 6 wherein parameters associated with each user entity are stored on the server preparatory to each user entity participating in the system.

8. The method of any one of claims 3 to 7 wherein parameters associated with each digital item are stored on the server preparatory to each user entity participating in the system.

9. The method of any one of claims 3 to 8 wherein the digital item is stored on the server in encrypted form.

10. The method of any one of claims 3 to 9 wherein the digital item is stored on the server in encrypted form by reference to a document key.

11. A method of secure access control and transmission of digital items in a client server environment incorporating a first underlying methodology wherein servers forming part of the system never have enough information to decrypt the user entity content.

12. The method of claim 11 incorporating a second underlying methodology wherein all user entity content to which a user may have access is encrypted (directly or transitively) against their private key.

13. The method of claim 11 or 12 wherein the servers forming the part of the system never see this key in an unencrypted form.

14. The method of any one of claims 11 to 13 wherein the servers store the key encrypted using a key derived from the user's password.

15. The method of any one of claims 11 to 14 wherein the servers forming part of the system never see the user's password (or any keys derived from it).

16. The method of claim 15 wherein the user's password is never sent to the servers, even in encrypted form.

17. The method of any one of claims 11 to 16 wherein the user's password is verified at the client device level by using the derivative of the user's password to decrypt the user key.

18. The method of any one of claims 11 to 17 wherein systems implementing the method rely on an infrastructure which includes cryptographic keys.

19. The method of claim 18 wherein cryptographic keys are associated with the various elements making up the systems.

20. The method of any one of claims 11 to 19 wherein cryptographic keys are associated with:
each entity or user participating in the system;
each digital device used to participate in the system;
each digital item stored or transmitted within the system.

21. The method of any one of claims 11 to 20 wherein the cryptographic keys are utilised to encrypt and decrypt each digital item (typically a document) stored or transmitted within the system, and to determine which users may decrypt a digital item.

22. The method of any one of claims 1 to 21 further applied to a method for authentication of a user login request in order to permit an authenticated user session between the user and a server on a device of the user in a client/server system where the device and user are previously jointly authenticated with respect to the server;
the user having at least for a given user login: a user ID, a user password, a user private key and a user public key;

the device having at least a device ID, a device public key and a device private key;
the server in a user enrolment event for the user storing the public key of the user and a derivati ve of the user private key of the user against the user ID;
the server in a device enrolment event for the device storing the device public key against the device ID;
said method comprising authenticating login of the user in a login event by the user to the server by the following steps:
provided the server has enrolled the user login;
provided the server has enrolled the device;
the device signals to the server commencement of the login event by the user;
the server supplies to the device the derivative of the user private key stored against the user ID and a randomly created authentication challenge;
the device performs a comparison operation to determine if the password of the user is the same as the password used to create the derivative to encrypt the user private key, and thereby decrypts the user private key from the derivative;
if the comparison operation is positive the device signals to the server that the user login request has been authenticated by providing a cryptographic response to the authentication challenge which requires possession of the user private key, whereupon the server treats the login event as successful such that the user ID is now authenticated for the user session on the server using the device.

23. The method of claim 22 wherein the comparison operation compares the user password supplied during the login attempt with the user password used to encrypt the user private key to generate the value stored on the server.

24. The method of claim 22 or 23 wherein the derivative of the user private key is an encryption of the private key encrypted against a value derived from the user password.

25. A method of secure access control and transmission of digital items in a client server environment as claimed in any previous claim further incorporating a password recovery method utilising a recovery card, said password recovery method comprising
in a first card generation step
executed on an off-line device;

for each card the off-line device generates a card private key and
a card password and
a card ID and
a card public key (derived from the card private key);
the off-line device then encrypts the card private key using the card password so as to generate an encrypted card private key;
the off-line device then embosses and/or prints on the card so as to be readable by a user the card ID and the card password;
the system administrator may then upload to an online server (platform)
the card ID and
the encrypted card private key and
the card public key;
in a second user setup step the user utilises a device which has been authenticated to the online server;
the user previously registered to the server whereby the server has a user record for the user comprising user ID; user public key; encrypted user private key;
the user indicates that they wish to register a card, supplying the card ID from the card;
the device having the user ID, user password, user public key and user private key;
the device then transmits the card ID to the online server in order to receive the card public key for that card identified by that card ID from the server:
the device then encrypts the user private key against the card public key to generate a card encrypted user private key;
the device uploads the card encrypted user private key to the online server where it is referenced against a user record on the online server;
thereby to enable a (third) user password reset step before the user;
said reset step comprising
the user loads to the device the recovery card password and (associated with) the card ID;
the device receives from the online server the encrypted user private key and the encrypted card private key
the device decrypts the encrypted card private key by using the recovery card password provided to the device by the user so as to obtain a recovered card private key;

the device decrypts the card encrypted user private key by using the recovered card private key in order to thereby obtain a recovered user private key;
the device encrypts the recovered user private key against a new user password provided to the device by the user as part of the password reset step;
thereby to produce a new encrypted user private key;
the device uploads the new encrypted user private key to the online server whereby the new encrypted private key replaces the encrypted private key previously associated with their user record corresponding to their user ID.

26. A method of secure access control and transmission of digital items in a client server environment;
the method further including a method of password replacement for a user using a device to communicate with the server;
the user authenticated with the server using a user set of credentials including a user ID and user password and user private key and user public key;
the device authenticated with the server using a device a set of credentials including a device ID and device password and device private key and device public key;
the method reliant on creation of a recovery card having its own set of credentials including a recovery card ID, a recovery card password, a recovery card private key and a recovery card public key;
the method comprising associating a selected recovery card with the user on the server whereby subsequently the user may utilise the device and selected ones of the recovery card credentials to replace selected ones of the user credentials on the server.

27. A method for authentication of a user login request in order to permit an authenticated user session between the user and a server on a device of the user in a client/server system where the device and user are previously jointly authenticated with respect to the server;
the user having at least for a given user login: a user ID, a user password, a user private key and a user public key;
the device having at least a device ID, a device public key and a device private key;
the server in a user enrolment event for the user storing the public key of the user and a derivative of the user private key of the user against the user ID;

the server in a device enrolment event for the device storing the device public key against the device ID;
said method comprising authenticating login of the user in a login event by the user to the server by the following steps:
provided the server has enrolled the user login;
provided the server has enrolled the device;
the device signals to the server commencement of the login event by the user;
the server supplies to the device the derivative of the user private key stored against the user ID and a randomly created authentication challenge;
the device performs a comparison operation to determine if the password of the user is the same as the password used to create the derivative to encrypt the user private key, and thereby decrypts the user private key from the derivative;
if the comparison operation is positive the device signals to the server that the user login request has been authenticated by providing a cryptographic response to the authentication challenge which requires possession of the user private key, whereupon the server treats the login event as successful such that the user ID is now authenticated for the user session on the server using the device.

28. The method of claim 27 wherein the comparison operation compares the user password supplied during the login attempt with the user password used to encrypt the user private key to generate the value stored on the server.

29. The method of claim 27 or 28 wherein the derivative of the user private key is an encryption of the private key encrypted against a value derived from the user password.

30. A client server environment operating according to the method of any previous claim.

31. A device operating according to the method of any previous claim.

32. A digital item secured according to the method of any previous claim.

33. Media containing code which when executed by a processor performs the method of any previous claim.

34. A recovery card marked with indicia and operable according to the method of any previous claim.

35. A system for secure access control and transmission of digital items; the system including at least one server and at least one client device; the server including at least one server processor in communication with a server memory; the server memory storing code for execution by the server processor; the at least one client device including at least one device processor in communication with a device memory; the device memory storing code for execution by the device processor; the system further including a recovery card having its own set of recovery card credentials; the apparatus utilised to effect password replacement for a user using the at least one client device to communicate with the at least one server;
a user authenticated with the server using a user set of credentials including a user ID and user password and user private key and user public key;
the device authenticated with the server using a device set of credentials including a device ID and device password and device private key and device public key;
the system reliant on creation of a recovery card having its own set of credentials including a recovery card ID, a recovery card password, a recovery card private key and a recovery card public key;
the system associating a selected recovery card with the user on the server whereby subsequently the user may utilise the device and selected ones of the recovery card credentials to replace selected ones of the user credentials on the server.

36. A system for authentication of a user login request in order to permit an authenticated user session between the user and a server on a device of the user in a client/server system where the device and user are previously jointly authenticated with respect to the server; the system including at least one server and at least one client device; the server including at least one server processor in communication with a server memory; the server memory storing code for execution by the server processor; the at least one client device including at least one device processor in communication with a device memory; the device memory storing code for execution by the device processor; the user having at least for a given user

login: a user ID, a user password, a user private key and a user public key;
the device having at least a device ID, a device public key and a device private key;
the server in a user enrolment event for the user storing the public key of the user and a derivative of the user private key of the user against the user ID;
the server in a device enrolment event for the device storing the device public key against the device ID;
said method comprising authenticating login of the user in a login event by the user to the server by the following steps:
provided the server has enrolled the user login;
provided the server has enrolled the device;
the device signals to the server commencement of the login event by the user;
the server supplies to the device the derivative of the user private key stored against the user ID and a randomly created authentication challenge;
the device performs a comparison operation to determine if the password of the user is the same as the password used to create the derivative to encrypt the user private key, and thereby decrypts the user private key from the derivative;
if the comparison operation is positive the device signals to the server that the user login request has been authenticated by providing a cryptographic response to the authentication challenge which requires possession of the user private key, whereupon the server treats the login event as successful such that the user ID is now authenticated for the user session on the server using the device.

37. A system for secure access control and transmission of digital items in a client server environment wherein at least one server communicates with a plurality of participating user entities by means of participating digital devices in the form of digital communications devices in order to provide secure access control and transmission services for digital items created by, originated by or brought into the environment by ones of the plurality of participating user entities;
the server and the device executing steps whereby:
for each participating user entity participating in the environment;

each participating user entity is authenticated as between the entity and the at least one server for that user entity in a participating user entity authentication step;
each participating digital device used by the participating user entity is authenticated as between the digital device and the at least one server for that user entity in a participating digital device authentication step;
and wherein each participating user entity has a participating user entity encryption key;
the steps further including:
a sending (source) user entity enabling secure access for a receiving (target) user entity of the plurality of user entities to a designated (unencrypted) digital item which is recorded as under the custody of the sending (source) user entity;
said sending (source) user entity encrypting the designated digital item or the user entity encrypting a key by which the designated digital item has been encrypted so as to produce an encrypted designated digital item or an encrypted digital item key preparatory to the sending user entity enabling secure access to the designated digital item by the receiving (target) user;
said sending (source) user entity utilising a sending (source) user entity digital device which has been authenticated as between the server and the sending (source) user entity thereby to enable the designated receiving (target) user entity to decrypt the designated digital item;
said sending user entity having a sending user entity encryption key;
said designated receiving user entity having a designated receiving user entity encryption key;
access to the designated digital item effected by the sending user entity encrypting the designated digital item so as to produce the encrypted designated digital item by utilising the designated receiving user entity encryption key to encrypt the designated digital item or access to the designated digital item effected by the sending user entity encrypting the document key of the designated digital item so as to produce the encrypted document key of the designated digital item by utilising the designated receiving user entity encryption key to encrypt the document key of the designated digital item.

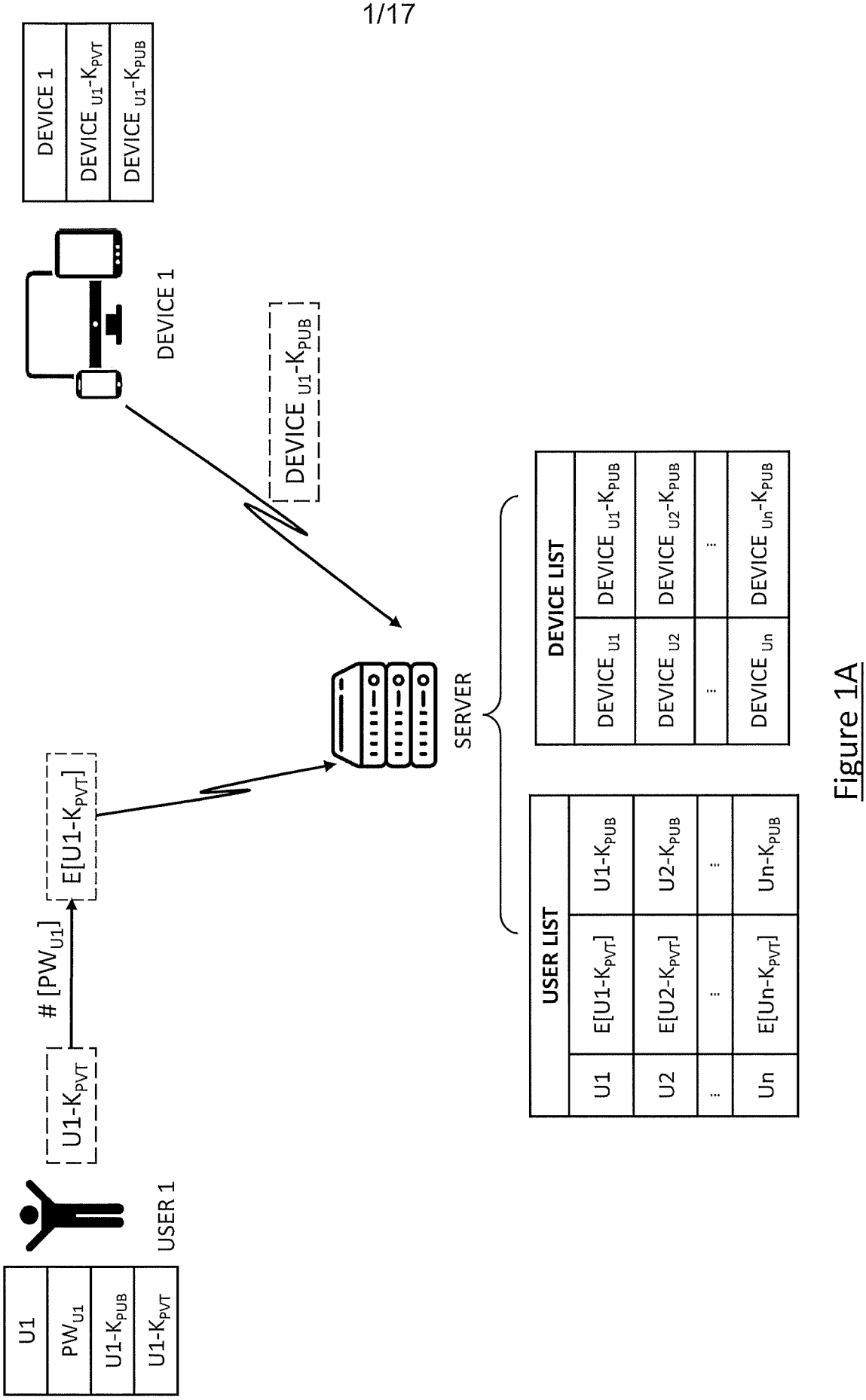

DEVICE 1
DEVICE U1-KPVT
DEVICE U1-KPUB
DEVICE 1
DEVICE U1-KPUB
SERVER
DEVICE LIST
DEVICE U1
DEVICE U1-KPUB
DEVICE U2
DEVICE U2-KPUB
DEVICE Un
DEVICE Un-KPUB
USER LIST
U1
E[U1-KPVT]
U1-KPUB
U2
E[U2-KPVT]
U2-KPUB
Un
E[Un-KPVT]
Un-KPUB
[PWU1]
E[U1-KPVT]
U1-KPVT
USER 1
U1
PWU1
U1-KPUB
U1-KPVT

Figure 1A

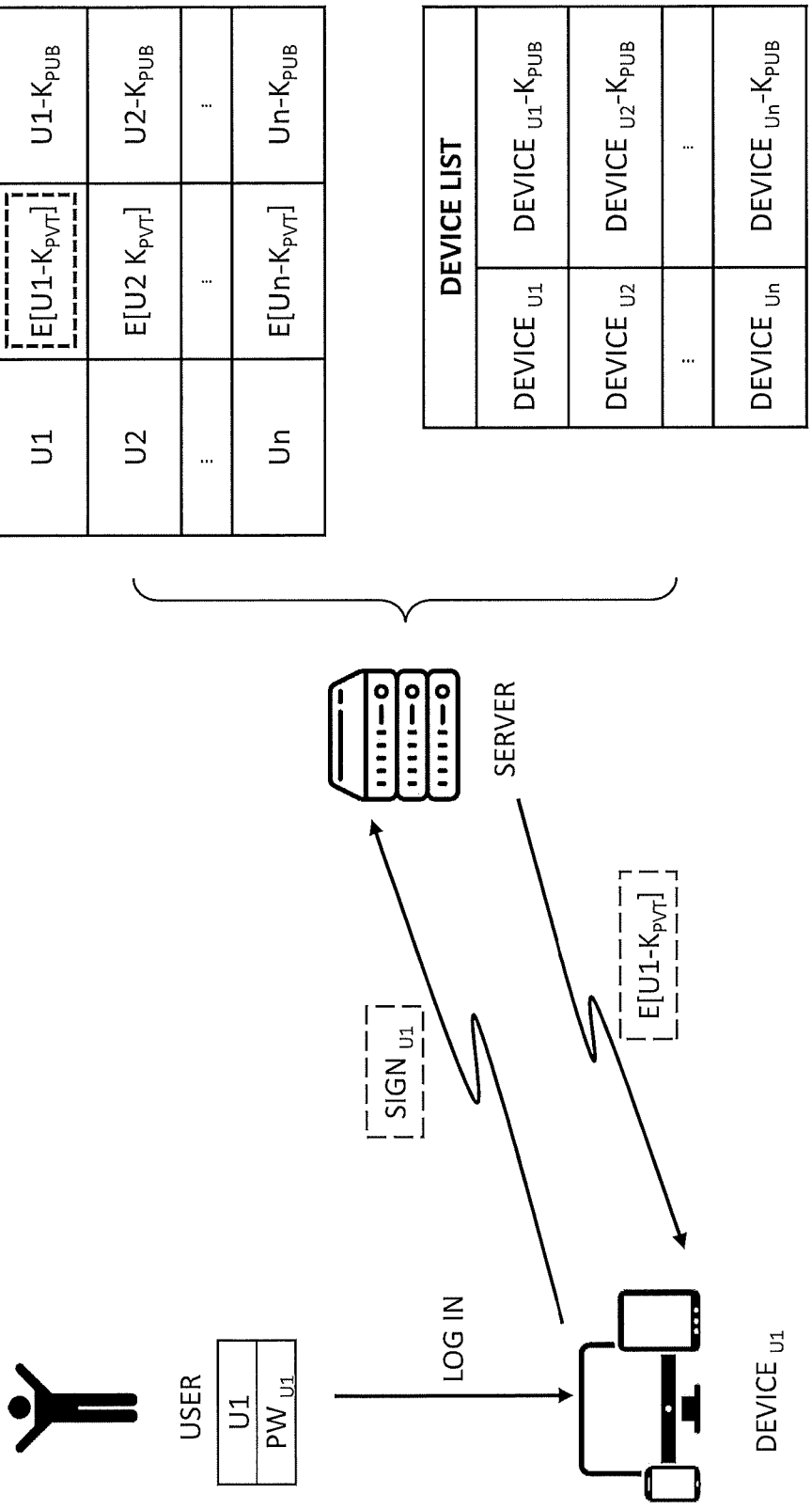

USER LIST
U1
U2
Un
E[U1-K_PVT]
E[U2 K_PVT]
E[Un-K_PUB]
U1-K_PUB
U2-K_PUB
Un-K_PUB
DEVICE LIST
DEVICE U1
DEVICE U2
DEVICE Un
DEVICE U1-K_PUB
DEVICE U2-K_PUB
DEVICE Un-K_PUB
SERVER
SIGN U1
E[U1-K_PVT]
LOG IN
USER
U1
PW U1
DEVICE U1

Figure 1B

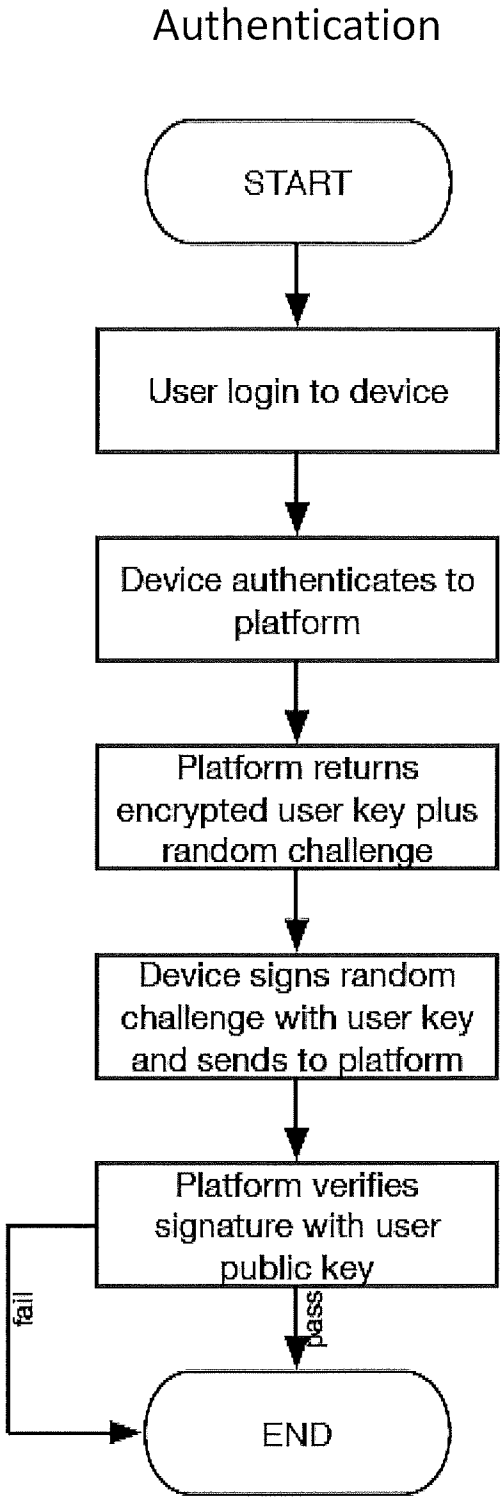

Authentication
START
User login to device
Device authenticates to platform
Platform returns encrypted user key plus random challenge
Device signs random challenge with user key and sends to platform
Platform verifies signature with user public key
fail
pass
END

Figure 2

U1
PW_U1
U1-K_PUB
U1-K_PVT
USER 1
DEVICE 1
[PW_U1]
E[U1-K_PVT]
U1-K_PVT
SERVER
USER LIST
U1
E[U1-K_PVT]
U1-K_PUB
U2
E[U2-K_PVT]
U2-K_PUB
Un
E[Un-K_PVT]
Un-K_PUB
RC1
PW_RC1
PIECE 1
RECOVERY CARD 1
[PW_RC1]
E[RC1-K_PVT]
RC1-K_PVT
RECOVERY CARD GENERATION SYSTEM
PIECE 2
RECOVERY CARD LIST
RC1
PW_RC1
RC1-K_PUB
RC1-K_Pvt
E[RC1-K_PVT]
RC2
PW_RC2
RC2-K_PUB
RC2-K_Pvt
E[RC2-K_PVT]
RCn
PW_RCn
RCn-K_PUB
RCn-K_Pvt
E[RCn-K_PVT]

Figure 3A

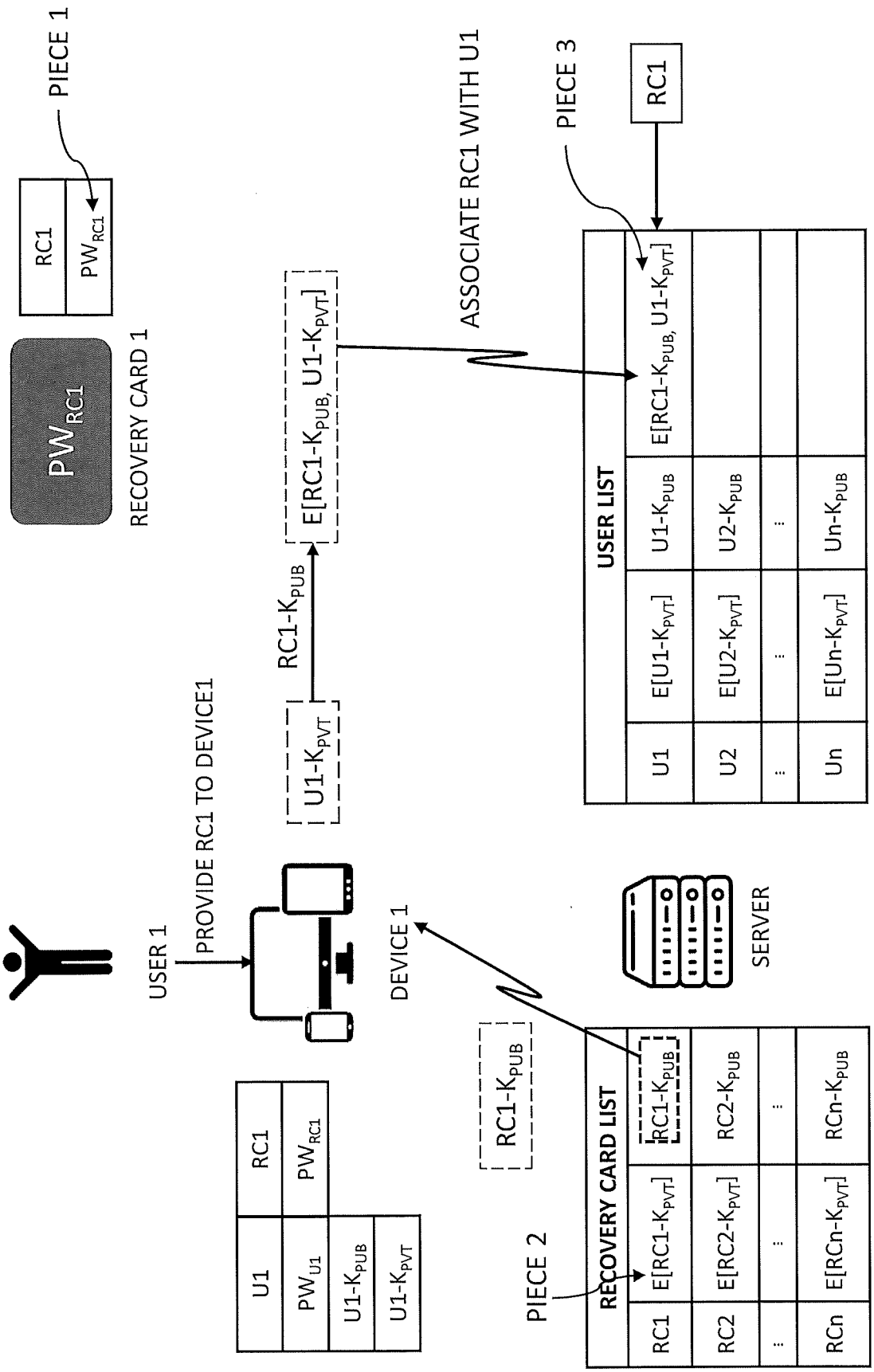
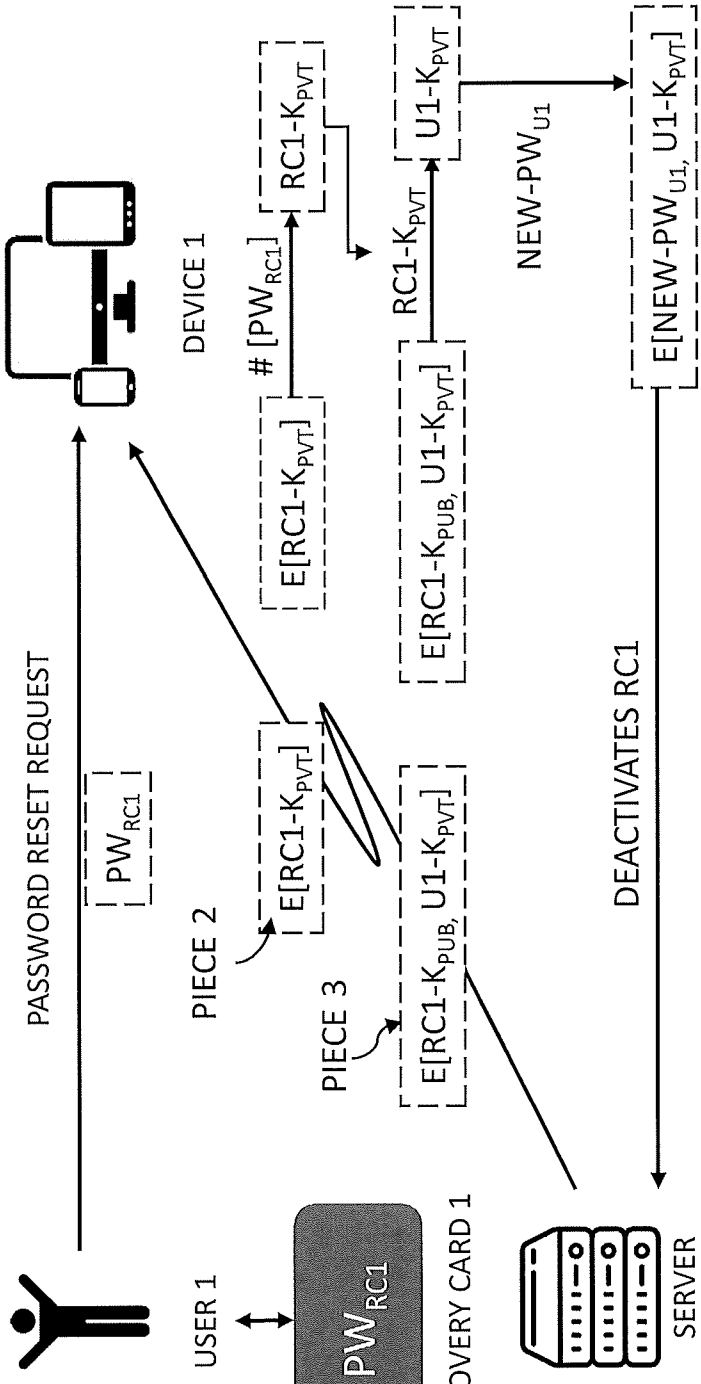

PIECE 1
RC1
PW$_{RC1}$
PW$_{RC1}$
RECOVERY CARD 1
ASSOCIATE RC1 WITH U1
PIECE 3
RC1
USER LIST
U1
E[U1-K$_{PVT}$]
U1-K$_{PUB}$
E[RC1-K$_{PUB}$, U1-K$_{PVT}$]
U2
E[U2-K$_{PVT}$]
U2-K$_{PUB}$
Un
E[Un-K$_{PVT}$]
Un-K$_{PUB}$
RC1-K$_{PUB}$
E[RC1-K$_{PUB}$, U1-K$_{PVT}$]
U1-K$_{PVT}$
USER 1
PROVIDE RC1 TO DEVICE1
DEVICE 1
SERVER
U1
PW$_{U1}$
U1-K$_{PUB}$
U1-K$_{PVT}$
RC1
PW$_{RC1}$
RC1-K$_{PUB}$
PIECE 2
RECOVERY CARD LIST
RC1
E[RC1-K$_{PVT}$]
RC1-K$_{PUB}$
RC2
E[RC2-K$_{PVT}$]
RC2-K$_{PUB}$
RCn
E[RCn-K$_{PVT}$]
RCn-K$_{PUB}$

Figure 3B



USER LIST

U1	E[U1-K$_{PVT}$]	U1-K$_{PUB}$	E[RC1-K$_{PUB}$, U1-K$_{PVT}$]
U2	E[U2-K$_{PVT}$]	U2-K$_{PUB}$	E[RC2-K$_{PUB}$, U2-K$_{PVT}$]
⋮	⋮	⋮	⋮
Un	E[Un-K$_{PVT}$]	Un-K$_{PUB}$	E[RCn-K$_{PUB}$, Un-K$_{PVT}$]

RECOVERY CARD LIST

RC1	E[RC1-K$_{PVT}$]	RC1-K$_{PUB}$
RC2	E[RC2-K$_{PVT}$]	RC2-K$_{PUB}$
⋮	⋮	⋮
RCn	E[RCn-K$_{PVT}$]	RCn-K$_{PUB}$

Figure 4

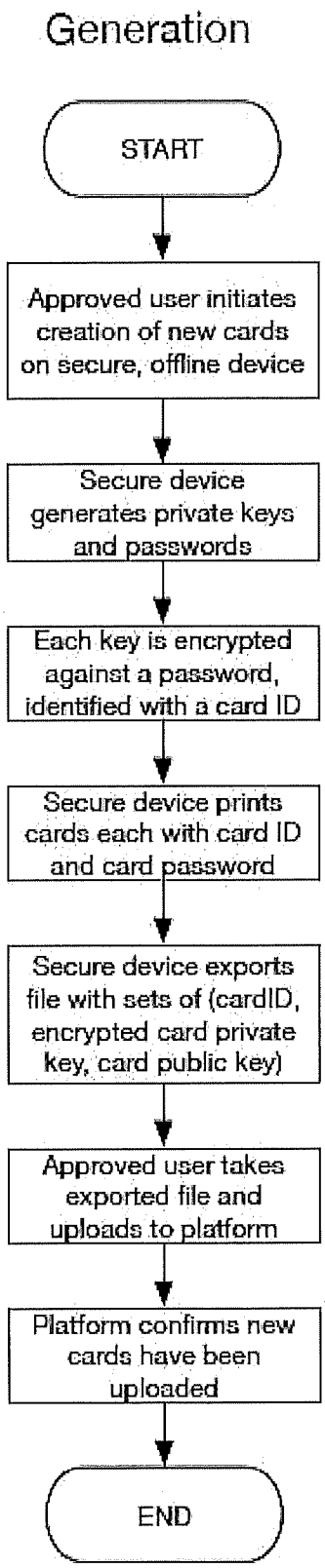

Generation
START
Approved user initiates creation of new cards on secure, offline device
Secure device generates private keys and passwords
Each key is encrypted against a password, identified with a card ID
Secure device prints cards each with card ID and card password
Secure device exports file with sets of (cardID, encrypted card private key, card public key)
Approved user takes exported file and uploads to platform
Platform confirms new cards have been uploaded
END

Figure 5A

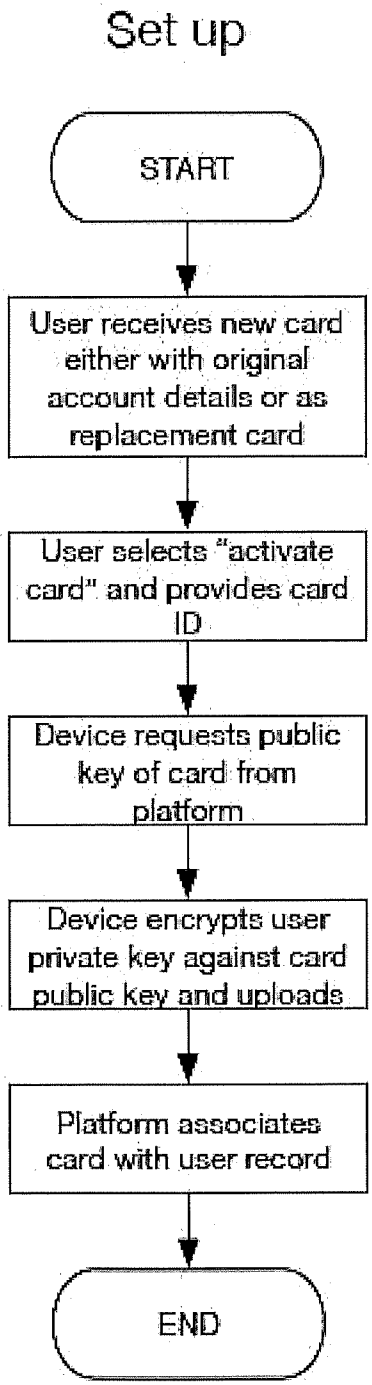

Set up
START
User receives new card either with original account details or as replacement card
User selects "activate card" and provides card ID
Device requests public key of card from platform
Device encrypts user private key against card public key and uploads
Platform associates card with user record
END

Figure 5B

Reset

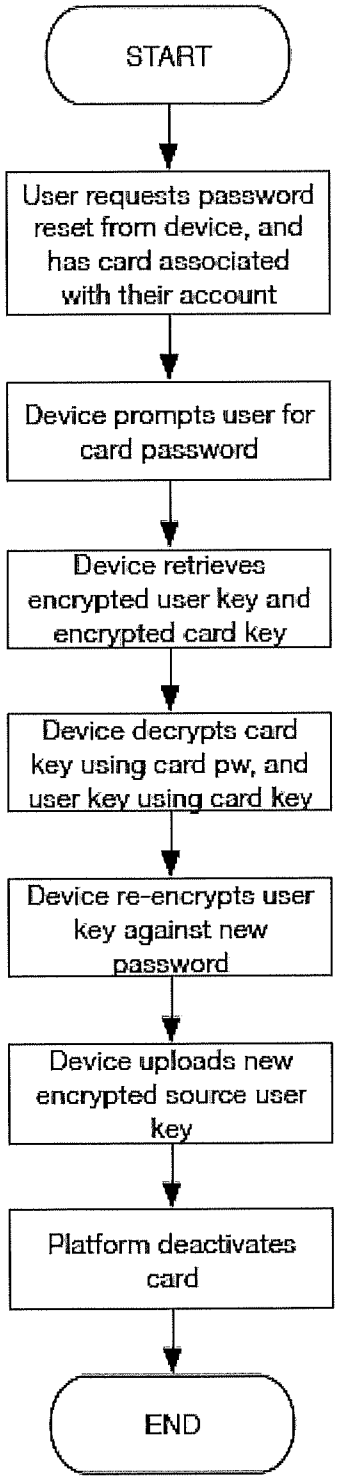


Figure 5C

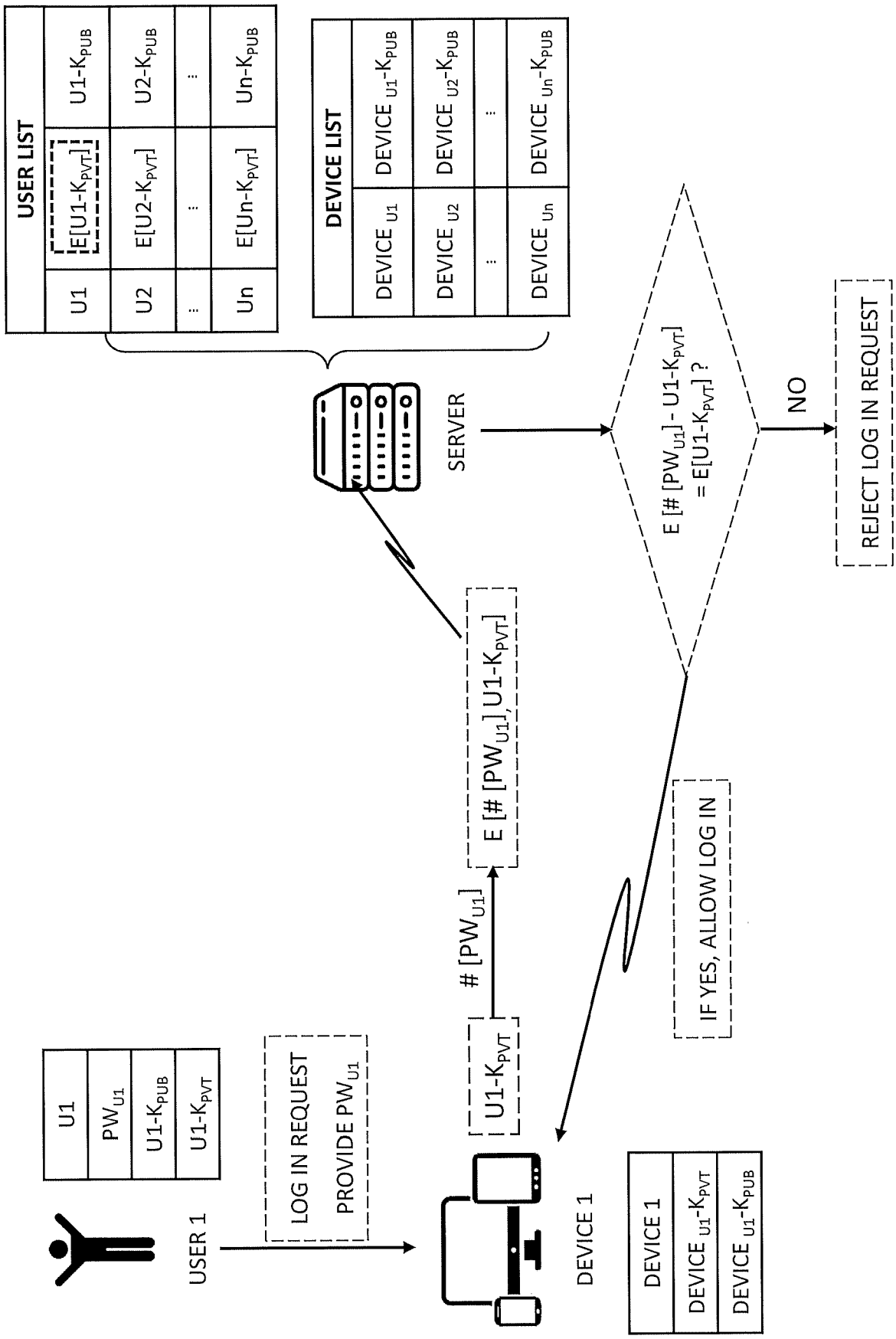

USER LIST
U1
E[U1-K_PVT]
U1-K_PUB
U2
E[U2-K_PVT]
U2-K_PUB
Un
E[Un-K_PVT]
Un-K_PUB
DEVICE LIST
DEVICE U1
DEVICE U1-K_PUB
DEVICE U2
DEVICE U2-K_PUB
DEVICE Un
DEVICE Un-K_PUB
SERVER
E [# [PW_U1] - U1-K_PVT] = E[U1-K_PVT] ?
NO
REJECT LOG IN REQUEST
U1
PW_U1
U1-K_PUB
U1-K_PVT
USER 1
LOG IN REQUEST
PROVIDE PW_U1
U1-K_PVT
[PW_U1]
E [# [PW_U1], U1-K_PVT]
IF YES, ALLOW LOG IN
DEVICE 1
DEVICE 1
DEVICE U1-K_PVT
DEVICE U1-K_PUB

Figure 6

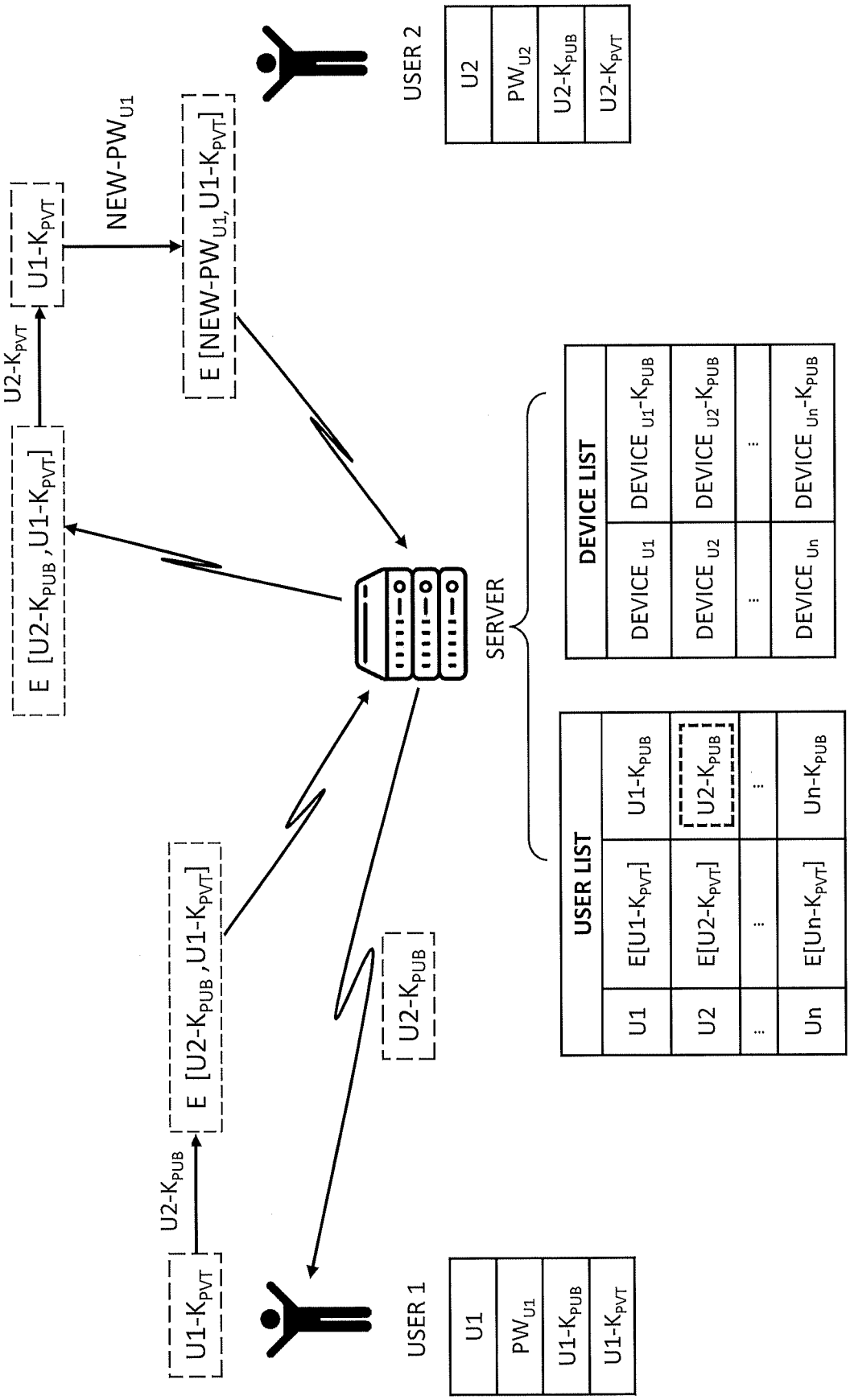

USER 2
U2
PW U2
U2-K PUB
U2-K PVT
NEW-PW U1
U1-K PVT
E [NEW-PW U1, U1-K PVT]
U2-K PVT
E [U2-K PUB , U1-K PVT]
SERVER
DEVICE LIST
DEVICE U1
DEVICE U1-K PUB
DEVICE U2
DEVICE U2-K PUB
DEVICE Un
DEVICE Un-K PUB
USER LIST
U1
E[U1-K PVT]
U1-K PUB
U2
E[U2-K PVT]
U2-K PUB
Un
E[Un-K PVT]
Un-K PUB
U2-K PUB
E [U2-K PUB , U1-K PVT]
U2-K PUB
U1-K PVT
USER 1
U1
PW U1
U1-K PUB
U1-K PVT

Figure 7

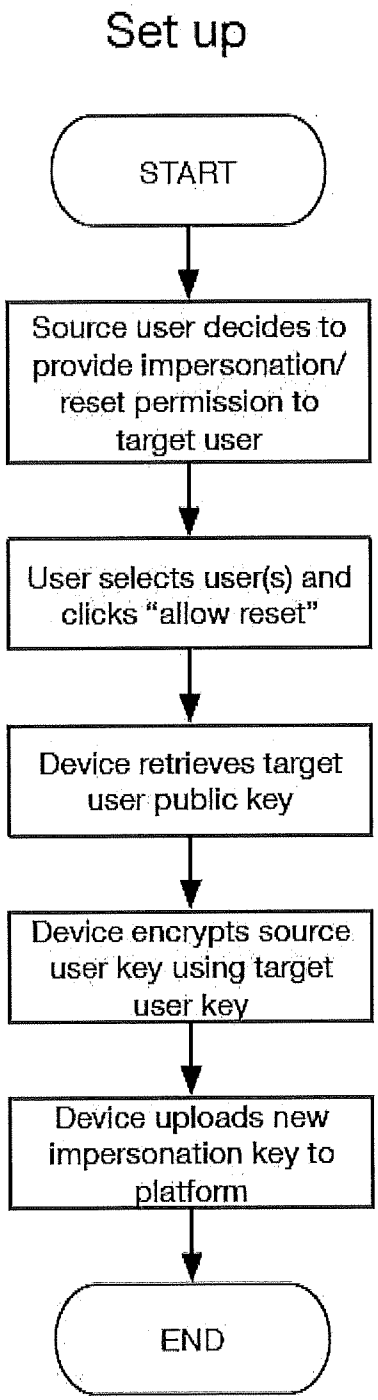

Set up
START
Source user decides to provide impersonation/ reset permission to target user
User selects user(s) and clicks "allow reset"
Device retrieves target user public key
Device encrypts source user key using target user key
Device uploads new impersonation key to platform
END

Figure 8A

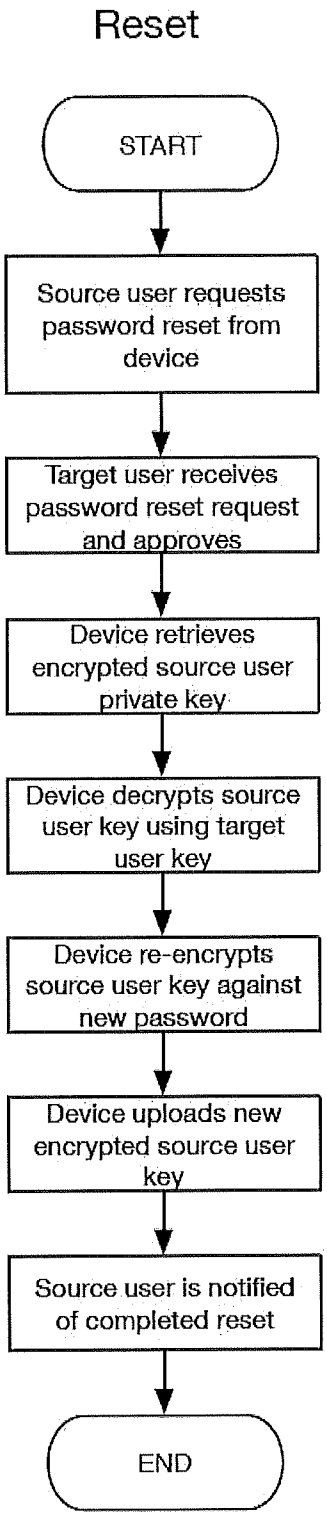

Reset
START
Source user requests password reset from device
Target user receives password reset request and approves
Device retrieves encrypted source user private key
Device decrypts source user key using target user key
Device re-encrypts source user key against new password
Device uploads new encrypted source user key
Source user is notified of completed reset
END

Figure 8B

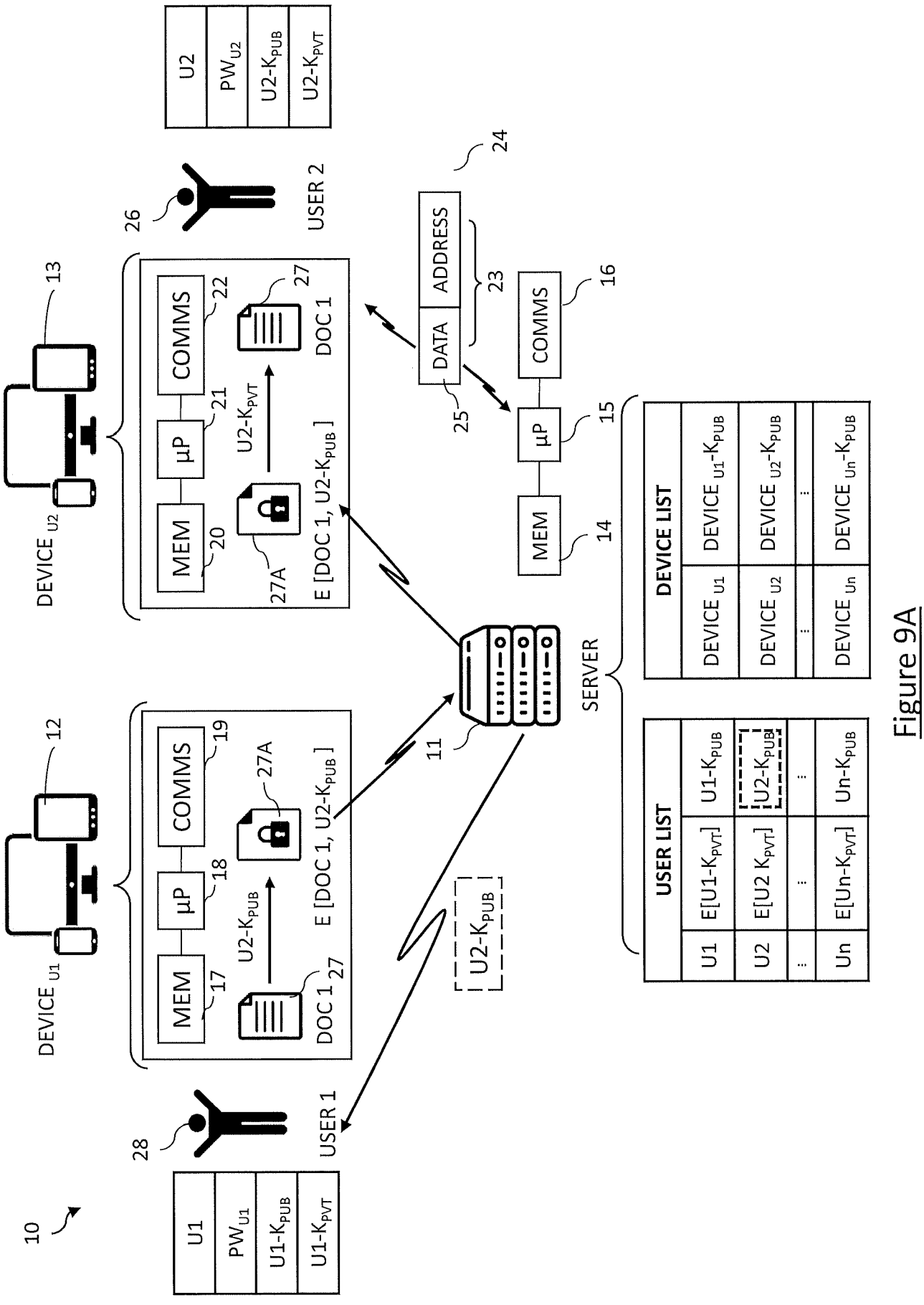

10
U1
PW U1
U1-K PUB
U1-K PVT
28
USER 1
DEVICE U1
12
MEM
17
µP
18
COMMS
19
U2-K PUB
27A
DOC 1 27
E [DOC 1, U2-K PUB]
U2-K PUB
11
SERVER
USER LIST
U1
E[U1-K PVT]
U1-K PUB
U2
E[U2 K PVT]
U2-K PUB
Un
E[Un-K PVT]
Un-K PUB
DEVICE LIST
DEVICE U1
DEVICE U1-K PUB
DEVICE U2
DEVICE U2-K PUB
DEVICE Un
DEVICE Un-K PUB
MEM
14
µP
15
COMMS
16
DATA
ADDRESS
23
24
25
DEVICE U2
13
MEM
20
µP
21
COMMS
22
U2-K PVT
27A
27
DOC 1
E [DOC 1, U2-K PUB]
26
USER 2
U2
PW U2
U2-K PUB
U2-K PVT

Figure 9A

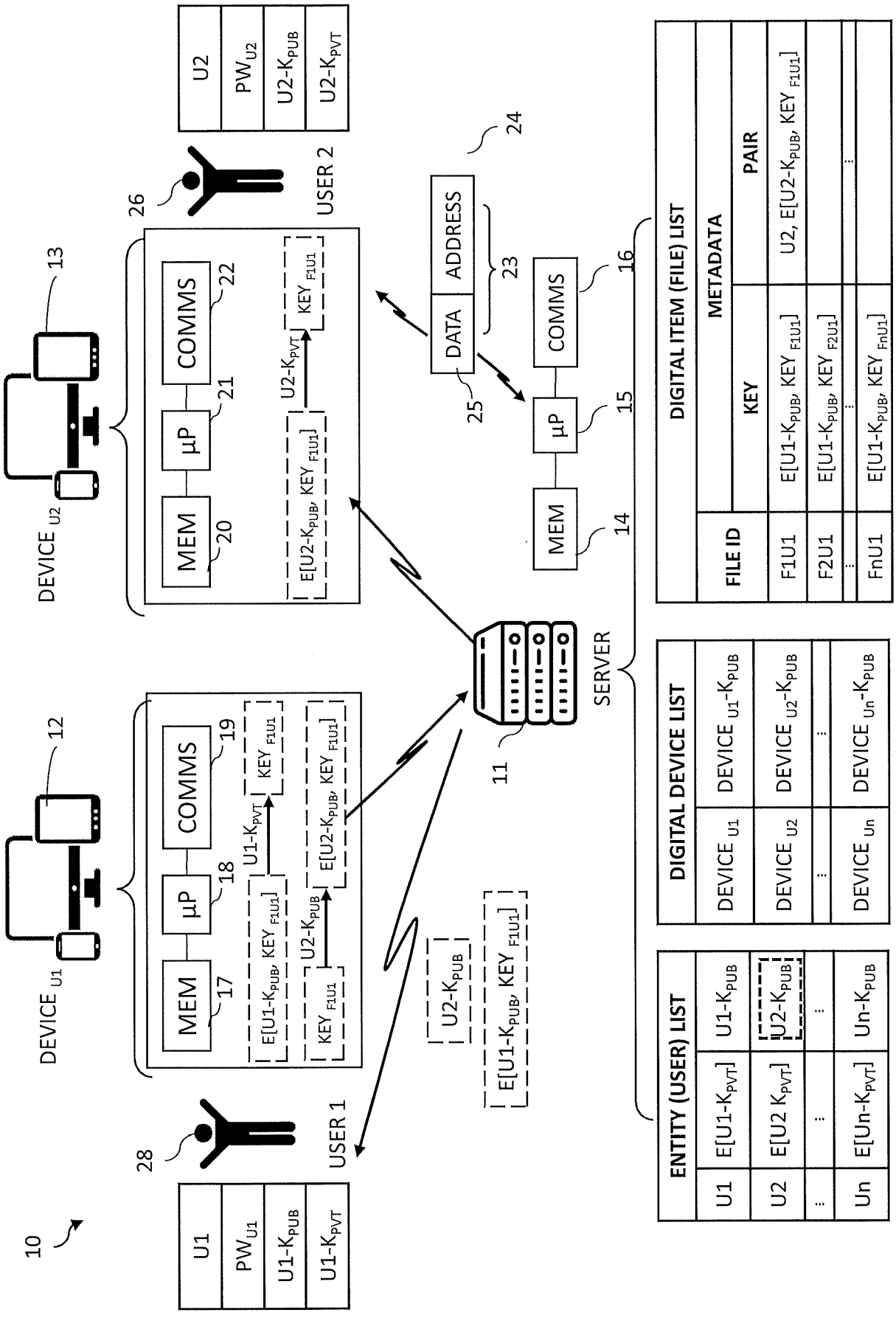

10
U1
PW U1
U1-K PUB
U1-K PVT
28
USER 1
DEVICE U1
12
MEM
17
µP
18
COMMS
19
E[U1-K PUB, KEY F1U1]
U1-K PVT
KEY F1U1
KEY F1U1
U2-K PUB
E[U2-K PUB, KEY F1U1]
U2-K PUB
E[U1-K PUB, KEY F1U1]
11
SERVER
MEM
14
µP
15
COMMS
16
DATA
ADDRESS
23
24
25
DEVICE U2
13
MEM
20
µP
21
COMMS
22
E[U2-K PUB, KEY F1U1]
U2-K PVT
KEY F1U1
26
USER 2
U2
PW U2
U2-K PUB
U2-K PVT
ENTITY (USER) LIST
U1 E[U1-K PVT] U1-K PUB
U2 E[U2 K PVT] U2-K PUB
Un E[Un-K PVT] Un-K PUB
DIGITAL DEVICE LIST
DEVICE U1 DEVICE U1-K PUB
DEVICE U2 DEVICE U2-K PUB
DEVICE Un DEVICE Un-K PUB
DIGITAL ITEM (FILE) LIST
FILE ID
METADATA
KEY
PAIR
F1U1 E[U1-K PUB, KEY F1U1] U2, E[U2-K PUB, KEY F1U1]
F2U1 E[U1-K PUB, KEY F2U1]
FnU1 E[U1-K PUB, KEY FnU1]

Figure 9B

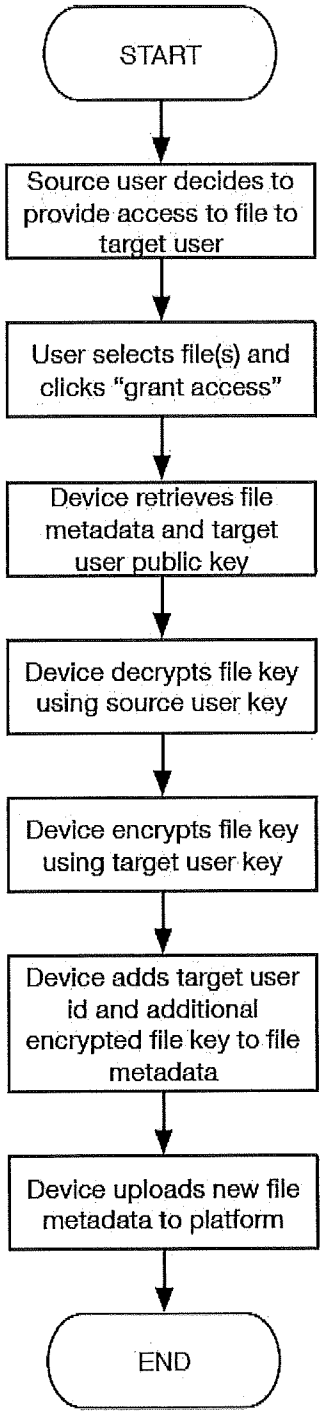

START
Source user decides to provide access to file to target user
User selects file(s) and clicks "grant access"
Device retrieves file metadata and target user public key
Device decrypts file key using source user key
Device encrypts file key using target user key
Device adds target user id and additional encrypted file key to file metadata
Device uploads new file metadata to platform
END

Figure 10

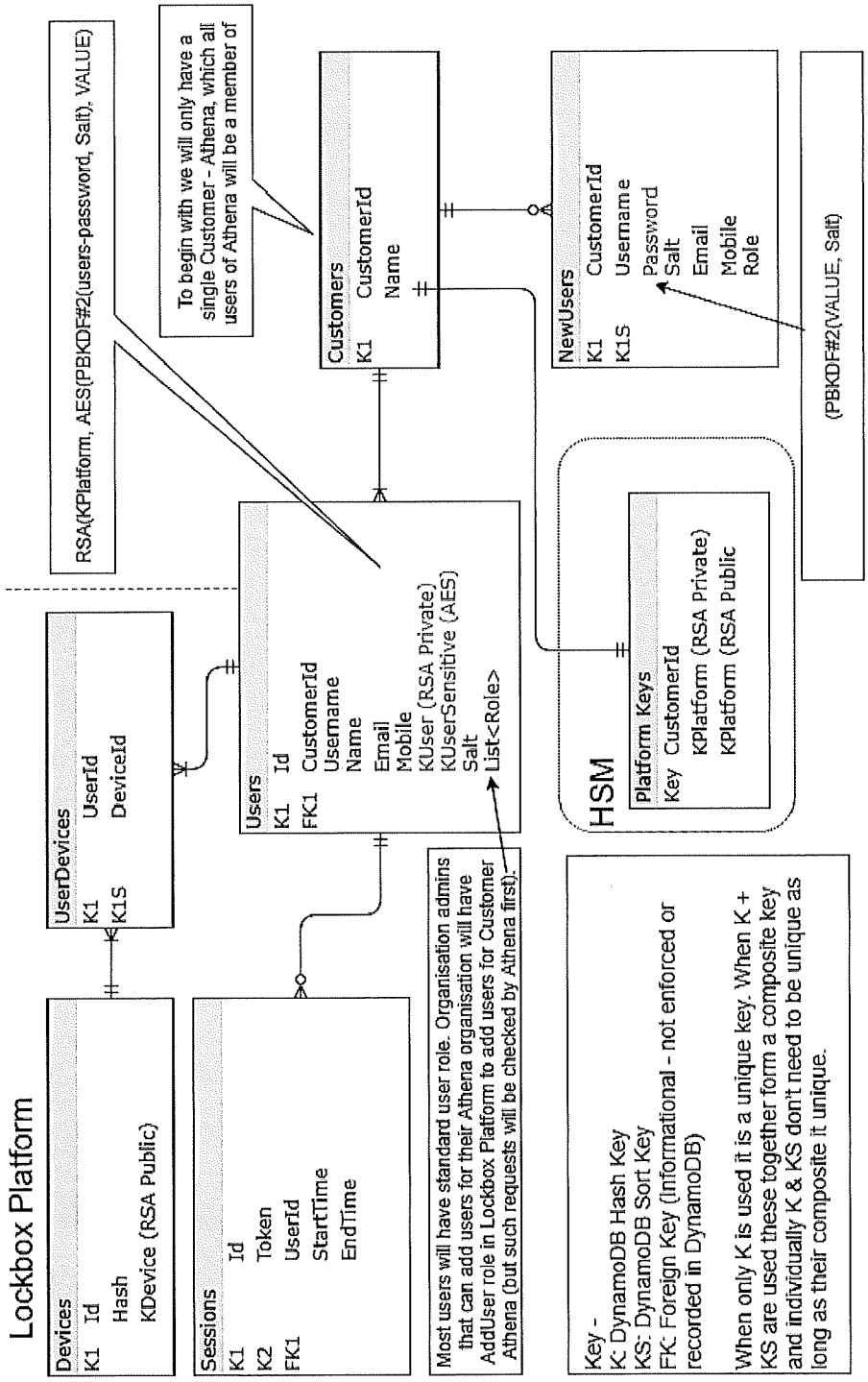

Lockbox Platform
Devices
K1 Id
Hash
KDevice (RSA Public)
UserDevices
K1 UserId
K1S DeviceId
Sessions
K1 Id
K2 Token
FK1 UserId
StartTime
EndTime
Users
K1 Id
FK1 CustomerId
Username
Name
Email
Mobile
KUser (RSA Private)
KUserSensitive (AES)
Salt
List<Role>
Most users will have standard user role. Organisation admins that can add users for their Athena organisation will have AddUser role in Lockbox Platform to add users for Customer Athena (but such requests will be checked by Athena first).
RSA(KPlatform, AES(PBKDF#2(users-password, Salt), VALUE)
To begin with we will only have a single Customer - Athena, which all users of Athena will be a member of
Customers
K1 CustomerId
Name
NewUsers
K1 CustomerId
K1S Username
Password
Salt
Email
Mobile
Role
(PBKDF#2(VALUE, Salt)
HSM
Platform Keys
Key CustomerId
KPlatform (RSA Private)
KPlatform (RSA Public
Key -
K: DynamoDB Hash Key
KS: DynamoDB Sort Key
FK: Foreign Key (Informational - not enforced or recorded in DynamoDB)
When only K is used it is a unique key. When K + KS are used these together form a composite key and individually K & KS don't need to be unique as long as their composite it unique.

Figure 11

INTERNATIONAL SEARCH REPORT

International application No.
PCT/AU2019/051138

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/30 (2013.01) H04L 9/08 (2006.01) H04L 9/32 (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)

PATENW: CPC G06F2221/2131, H04L63/08; IPC & CPC G06F21/30, H04L9/08, H04L9/32 & Keywords (card, carrier, emboss, print, stamp, password, passcode, passphrase, recover, reset, encrypt, encode, private, public, key, share, create, originate, source, digital, item, document, file, user, entity, identity, token) & like terms; GOOGLE PATENTS: Keywords (card, print, password, recover, reset, document, key) & like terms; AUSPAT, PATENTSCOPE, ESPACENET, IP Australia Internal Databases: Applicant/inventor name search

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	Documents are listed in the continuation of Box C	

[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

"D" document cited by the applicant in the international application

"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
29 January 2020	29 January 2020
Name and mailing address of the ISA/AU	**Authorised officer**
AUSTRALIAN PATENT OFFICE PO BOX 200, WODEN ACT 2606, AUSTRALIA Email address: pct@ipaustralia.gov.au	Simon Ellis AUSTRALIAN PATENT OFFICE (ISO 9001 Quality Certified Service) Telephone No. +61262832673

INTERNATIONAL SEARCH REPORT C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		International application No. **PCT/AU2019/051138**
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X Y	WO 2017/023385 A2 (PRIVATE MACHINES INC.) 09 February 2017 abstract, paras [00013], [00016], [00096]-[000120] and [000138]-[000149] abstract, paras [00013], [00016], [00096]-[000120] and [000138]-[000149]	3-24, 27-34, 36-37 1-2. 25-26, 35
X	WO 2016/060568 A1 (INVENIA AS) 21 April 2016 abstract, Figs. 16, 20B, pages 4-6, 14-24	3-24, 27-34, 36-37
X	US 2015312227 A1 (ADOBE SYSTEMS INCORPORATED) 29 October 2015 abstract, Figs. 1-2, paras [0026]-[0028], [0039]-[0064]	3-24, 27-34, 36-37
X	WO 2012144909 A1 (INVENIA AS) 26 October 2012 abstract, pages 1-7	3-24, 27-34, 36-37
X	US 2006/0075228 A1 (BLACK et al.) 06 April 2006 abstract, paras [0089]-[0119]	3-24, 27-34, 36-37
Y	CN 106685645 A (ZHENGZHOU XINDA JIEAN INFORMATION TECH CO LTD) 17 May 2017 & Machine English language translation sourced from Google Patents viewed abstract	1-2. 25-26, 35
Y	US 2013/0159699 A1 (F-SECURE CORPORATION) 20 June 2013 paras [0083]-[0087]	1-2, 25-26, 35

INTERNATIONAL SEARCH REPORT

International application No.
PCT/AU2019/051138

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:

because they relate to subject matter not required to be searched by this Authority, namely:

the subject matter listed in Rule 39 on which, under Article 17(2)(a)(i), an international search is not required to be carried out, including

2. ☐ Claims Nos.:

because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos:

because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a)

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

See Supplemental Box for Details

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☒ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.

☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.

☐ No protest accompanied the payment of additional search fees.

Supplemental Box

Continuation of: **Box III**

This International Application does not comply with the requirements of unity of invention because it does not relate to one invention or to a group of inventions so linked as to form a single general inventive concept.

This Authority has found that there are different inventions based on the following features that separate the claims into distinct groups:

- Claims 1-2, 25-26, 35 (and 22-24, 30-34 in part) are directed to a password recovery/replacement method and a system for secure access control and transmission of digital items.. The feature of a recovery card having its own credentials is specific to this group of claims.

- Claims 3-10, 37 (and 22-24, 30-34 in part) are directed to a method of secure access and transmission of digital items and a system for secure access control and transmission of digital items. The feature of exchanging encrypted digital items between a sender and a receiver where a server authenticates both parties and employing an encrypted document key is specific to this group of claims.

- Claims 11-21 (and 22-24, 30-34 in part) are directed to a method of secure access and transmission of digital items . The feature of secure access control and transmission of digital items in a client server environment where servers never have enough information to decrypt the user entity content is specific to this group of claims.

- Claims 27-29, 36 (and 30-34 in part) are directed to a method of authenticating a user login request and a system for authenticating a user login request. The feature of authenticating a user employing a derivative of the user private key against the user ID is specific to this group of claims.

PCT Rule 13.2, first sentence, states that unity of invention is only fulfilled when there is a technical relationship among the claimed inventions involving one or more of the same or corresponding special technical features. PCT Rule 13.2, second sentence, defines a special technical feature as a feature which makes a contribution over the prior art.

When there is no special technical feature common to all the claimed inventions there is no unity of invention.

In the above groups of claims, the identified features may have the potential to make a contribution over the prior art but are not common to all the claimed inventions and therefore cannot provide the required technical relationship. Therefore there is no special technical feature common to all the claimed inventions and the requirements for unity of invention are consequently not satisfied *a priori*.

INTERNATIONAL SEARCH REPORT Information on patent family members	International application No. **PCT/AU2019/051138**

This Annex lists known 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/s Cited in Search Report		**Patent Family Member/s**	
Publication Number	**Publication Date**	**Publication Number**	**Publication Date**
WO 2017/023385 A2	09 February 2017	WO 2017023385 A2	09 Feb 2017
		EP 3320440 A2	16 May 2018
		EP 3320447 A2	16 May 2018
		US 2019087432 A1	21 Mar 2019
		US 2019087600 A1	21 Mar 2019
		WO 2017023388 A2	09 Feb 2017
WO 2016/060568 A1	21 April 2016	WO 2016060568 A1	21 Apr 2016
		EP 3207725 A1	23 Aug 2017
		US 2017244556 A1	24 Aug 2017
US 2015312227 A1	29 October 2015	US 2015312227 A1	29 Oct 2015
		US 9521001 B2	13 Dec 2016
		US 2017032112 A1	02 Feb 2017
		US 9842201 B2	12 Dec 2017
WO 2012144909 A1	26 October 2012	WO 2012144909 A1	26 Oct 2012
		EP 2839407 A1	25 Feb 2015
		EP 2839407 B1	05 Sep 2018
		US 2015113279 A1	23 Apr 2015
		US 9224003 B2	29 Dec 2015
		US 2015161410 A1	11 Jun 2015
		US 9582678 B2	28 Feb 2017
		WO 2013157957 A1	24 Oct 2013
US 2006/0075228 A1	06 April 2006	US 2006075228 A1	06 Apr 2006
		US 2006005017 A1	05 Jan 2006
CN 106685645 A	17 May 2017	CN 106685645 A	17 May 2017
		CN 106685645 B	28 May 2019
US 2013/0159699 A1	20 June 2013	US 2013159699 A1	20 Jun 2013
		GB 2498039 A	03 Jul 2013
		GB 2498039 B	13 Nov 2013

End of Annex

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