(54) Title: IMPROVED STORAGE SYSTEM
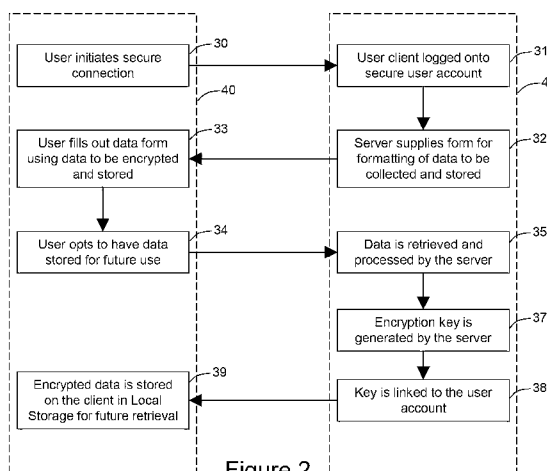
Figure 2

(57) Abstract: In a client server environment a method of, and apparatus for, securely storing data, said method comprising: generating a data element at a second location; transmitting the data element to a first location separate and remote from the second location; encrypting the data element at the first location, using a key, thereby to form an encrypted data element; transmitting the encrypted data element to the second location separate and remote from the first location and storing the encrypted data element at the second location; and wherein the second location is constituted as a client device.

Improved Storage System

Technical Field

[0001]   The present invention relates to the storage of data
on electronic networks and, more particularly although not
exclusively, to the secure storage of data which is accessible
across an electronic network.

Background

[0002]   A common practice is to secure sensitive or secure
data on a centralised server is a common practice in the art.
In the past this was an efficient way of storing data in that
the security resources of a server were considered more secure
and efficient than trying to store the same data on the client
side device.

[0003]   However the advent of more sophisticated hacking
attacks and the inherent danger of having lots of sensitive
stored data in one location for many people means that the
technical, legal and business risk of storing sensitive data
at a centralised location is becoming a large liability.

[0004]   Conversely storing encrypted data on a client side
device, while distributing the risk and liability, also
presents a security issue in that typically the client device
must also be used for the storage of the decryption key or the
user must go through the inconvenience of storing and
retrieving a decryption key from a handy nearby source. It is
imperative in a secure data system to store the encrypted data
and the decryption key in separate and secure locations, yet
the current measures to achieve this are typically complicated
and inconvenient for the user.

Prior Art

[0005]   PCT/AU2012/000944 [WO 2013020178] [Nussbaum et al]
assigned to Cocoon Data Holdings Limited discloses a
methodology which, broadly, seeks to remove the need for
management of encryption or decryption keys at the user level
or on the user device. In this citation this management
function is devolved to a separate server device which relies
on authentic of relevant parties before divulging or utilising
a decryption key. At page 19 lines 4 to 10 in part it is
stated "a further advantage is that there is no footprint
left… no software or secure data is installed or maintained on
the executives (which is to say client) computer…" (Underlined
words added). There is no teaching of storing the file secured
by the process on the user (client) device.

[0006]   US2007/0198823 [Blew et al] assigned to Educational
Testing Service (see also US7519810) also has as a major
objective to shield an end user from the mechanics of
encryption and decryption including shielding the end user
from the need to be aware of any encryption or decryption
keys. In this citation encryption and decryption and the
securing of a file are carried out on a server (in preferred
forms by a methodology requiring data transfer between
multiple servers) clearly separated from a client device and
there is no teaching of storing the file secured by the
process on the client device.

[0007]   Embodiments of the disclosed invention are designed
to address these issues.

Notes

[0008]   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".

[0009]   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.

Brief description of invention

Definitions:
[00010]   **Client server environment**: an arrangement of computing resources having a topology wherein client devices communicate over a network with at least logically and in most instances usually physically separate server devices. Broadly the server devices will usually have greater resources than the client devices and the client devices rely, at least in part, on communication with one or more of the server devices to carry out tasks on the client device.

[00011]   **Encryption/decryption**: in relation to a file containing data a methodology involving applying an algorithm such that the data is reformatted in a way which renders it substantially indecipherable unless a reverse algorithm is then applied to the data so as to return it to its unencrypted form. In preferred forms a "key" is used as a reference element in the algorithm. In some forms the same key is used to encrypt the data as to decrypt the data. In other forms the key used for encryption is different to the key used for decryption. Encryption is to be distinguished from simple password protection where access to the data within a file requires a password but the data itself within the file is not reformatted.

4

[00012]   **Authentication**: authentication in this specification is a methodology whereby the identity of an entity is ascertained to a predetermined level of certainty-usually as a precursor to permitting execution of functionality specific to, related to, authorised by or permitted by that entity. In its simplest form authentication is often carried out by a combination of a username and a password-that is the provision of a given username with a corresponding password pair comprises authentication. The entity may be an individual operating software on a particular platform or the entity may be the platform itself. By way of example client devices such as a smart phone may qualify as an entity. A given user of the smart phone may also qualify as an entity as an alternative to or as well as the smart phone.

[00013]   Accordingly in one broad form of the invention there is provided in a client server environment a method of securely storing data; said method comprising

[00014]   generating a data element at a first location

[00015]   encrypting the data element at the first location thereby to form an encrypted data element

[00016]   transmitting the encrypted data element to a second location remote from the first location

[00017]   storing the encrypted data element at the second location.

[00018]   In a further broad form of the invention there is provided in a client server environment a method of securely storing data; said method comprising:

[00019]   generating a data element at a second location; transmitting the data element to a first location separate and remote from the second location;

[00020]   encrypting the data element at the first location thereby to form an encrypted data element;

[00021]   transmitting the encrypted data element to the second location separate and remote from the first location and

[00022]   storing the encrypted data element at the second location; and wherein the second location is constituted as a client device.

[00023]   Preferably a decryption key for the encrypted data element is stored at the first location.

[00024]   Preferably the second location is a client device.

[00025]   Preferably the client device is programmed to execute a Web enabled application which has a storage capability.

[00026]   Preferably the encrypted data element is stored on the client device utilising the storage capability of the Web enabled application.

[00027]   Preferably the Web enabled application is a Web browser.

[00028]   Preferably the web browser executes HTML 5 including the HTML 5 local storage function.

[00029]

[00030]   In a further broad form of the invention there is provided in a client server environment a method for storing data securely; said method comprising encrypting the data at a first location using a processor located at the first location; the data thus encrypted comprising encrypted data which requires a key to decrypt the encrypted data; the key stored at the first location; the encrypted data transmitted

to a second location remote from the first location and stored at the second location until further processing is required to be done on the data at which time the encrypted data is transmitted to the first location and unencrypted by applying the key to a decryption algorithm executed by the processor.

[00031]   In a further broad form of the invention there is provided in a client server environment an apparatus for secure storage of data; said apparatus comprising a first processor at a first location which encrypts data utilising a key; set apparatus further comprising a second processor located at a second location remote from the first location; the data, after encryption, moved over a network to the second processor and stored in association with the second processor until the data is required for execution of an application on the first processor at which time the data is moved back from the second processor to the first processor and the first processor applies the key to a decryption algorithm to decrypt the data for use by the application executing on the first processor.

[00032]   In a further broad form of the invention there is provided a method of isolation from its key of data encrypted by means of the key and whereby separate retrieval of the key by a user for decryption of the data is not required; said method comprising-, in a client/server environment, receiving data from the server during an unencrypted window period; working on the data on the client during the unencrypted window period; transmitting the data to the server at the end of the unencrypted window period; the server encrypting the data with reference to the key to form encrypted data; the server then transmitting the encrypted data to the client; the server then deleting all of the data and the encrypted data on

the server thereby terminating the unencrypted window period;
the key remaining stored only on the server.

[00033]   In a further broad form of the invention there is
provided a method of isolation from its decryption key of
encrypted data stored on a client device; said method
comprising
storing the decryption key referenced against a username
account/ user login-data for the client device on a separate
device;
retrieving the data by transferring the encrypted data to the
separate device and upon authentication of the user
login/username account decrypting the encrypted data using the
decryption key.

[00034]   Preferably the separate device is a Web server.

[00035]   Preferably the user login is a username and
password.

[00036]   In a further broad form of the invention there is
provided in a client server environment a secured non-use
window where data is stored securely on the client has
encrypted data requiring decryption by decrypting the
encrypted data and wherein the decryption key is not stored on
the client.

[00037]   Preferably the decryption key is stored on the
server.

[00038]   Preferably the encryption/decryption is not done on
the client.

[00039]   Preferably the encryption/decryption is done on the server.

[00040]   Preferably the decryption key is released for use if and only if authentication of a client/service session is validated.

[00041]   Preferably the environment requires authentication at the user level.

[00042]   Preferably decryption key is referenced against the authentication.

[00043]   Preferably authentication is at the user device level.

[00044]   Preferably the data is a small data quantum.
[00045]   Preferably the data quantum is 100 MB.

[00046]   Preferably the data quantum is 50 MB.

[00047]   Preferably the data quantum is 10 MB.

[00048]   Preferably the data quantum is 1 MB.

[00049]   Preferably the data quantum is 0.5 MB.

[00050]   Preferably the data quantum is 0.1 MB.

[00051]   Preferably the data is only changed incrementally, if at all, each user session.

[00052]   Preferably only some elements of the data are changed, if at all, each user session.

[00053]   In a further broad form of the invention there is provided a method of securely storing data; said method comprising generating a data element at a first location encrypting the data element at the first location thereby to form an encrypted data element transmitting the encrypted data element to a second location remote from the first location storing the encrypted data element at the second location.

[00054]   Preferably a decryption key for the encrypted data element is stored at the first location.

[00055]   Preferably the second location is a client device.

[00056]   Preferably the client device is programmed to execute a Web enabled application which has a storage capability.

[00057]   Preferably the web enabled application executes HTML 5 including the HTML 5 local storage function.

[00058]   Preferably the encrypted data element is stored on the client device utilising the storage capability of the Web enabled application.

[00059]   Preferably the Web enabled application is a Web browser.

[00060]   In a further broad form of the invention there is provided a method for storing data securely; said method comprising encrypting the data at a first location using a processor located at the first location; the data thus

encrypted comprising encrypted data which requires a key to decrypt the encrypted data; the key stored at the first location; the encrypted data transmitted to a second location remote from the first location and stored at the second location until further processing is required to be done on the data at which time the encrypted data is transmitted to the first location and unencrypted by applying the key to a decryption algorithm executed by the processor.

[00061]   In a further broad form of the invention there is provided an apparatus for secure storage of data; said apparatus comprising a first processor at a first location which encrypts data utilising a key; set apparatus further comprising a second processor located at a second location remote from the first location; the data, after encryption, moved over a network to the second processor and stored in association with the second processor until the data is required for execution of an application on the first processor at which time the data is moved back from the second processor to the first processor and the first processor applies the key to a decryption algorithm to decrypt the data for use by the application executing on the first processor.

Drawings

[00062]   Embodiments of the present invention will now be described with reference to the accompanying drawings wherein:

[00063]   Figure 1 - Main components of the example embodiment.

[00064]   Figure 2 - Control process for initial storage of data in accordance with an embodiment of the invention.

[00065]   Figure 3 - Control process for subsequent use of
encrypted data in accordance with an embodiment of the
invention.

[00066]   Figures 4A, B, C describe in diagrammatic form one
form of execution of an example embodiment of the invention.

[00067]   Figure 5 illustrates in block diagram form a
client/server environment suitable for use with any of the
described embodiments.

[00068]   Figures 6A-P illustrates steps in execution of an
example of an embodiment of the invention, and use.

Description and Operation

[00069]   Figure 1 describes the main components of an example
embodiment. In the example embodiment the client device 10
stores an encrypted version of the data being stored 11.
However the key to decrypt the data 18 is not stored on the
client 10 but is stored on a server 14 that is usually
accessible over a network 15 such as the Internet.

[00070]   The key 18 to decrypt the data 11 is stored on the
server 14 in association with a specific user account 17 in a
database of users 16 that have secure access to the server 14.

[00071]   The encrypted data 11 is stored in the domain
related Local Storage1 area 13 of a browser 12 or web enabled
application 12 running on the client device 10.

---

[1] Examples of Local Storage code can be seen here
http://www.w3schools.com/html/html5_webstorage.asp

12

[00072]   The server 14 is used for various processes associated with and involving the encryption2, collection and processing of the data which is subsequently stored on the client 10 as an encrypted data file 11.

[00073]   Figure 2 discloses the control process for the initial storage and encryption of data. The user 40 initiates a secure connection 30 to the server 41 and the client device is logged onto a secure user account 31. Subsequently the server 40 supplies a form for use by the user to format the data to be secured 32 and the user fills out the form with some or all of the data to be secured 33. The user then opts to store the data securely on their client device 34.

[00074]   The server then retrieves the entered data and processes it 35. Processing may or may not include data verification and integrity checking. Subsequently an encryption key is generated by the server 36 and the key is linked to the current user's account 37.

[00075]   The key is then used to encrypt the data to be protected 38 and the resulting encrypted data file or files is stored on the client device in the browser or web enabled applications Local Storage for future retrieval 39.

[00076]   Figure 3 shows a control process for the subsequent use of encrypted data. Initially the user starts a secure connection 50 and the server then establishes a secure connection using a user account 51.

[00077]   During the course of use the server may present the user with a form to collect information that the user has

---

2 Encryption could involve AES encryption algorithm explained at
https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

opted to encrypt and securely store on their device in the past 52. The user is then given the option and chooses to use the encrypted data already stored on their client device 53.

[00078]   Subsequently the server retrieves the encrypted data from the client Local Storage 54. The server then retrieves the decryption key from the associated username account 55 and the data is decrypted on the server to memory 56 and the data is processed as needed 57.

[00079]   Subsequently the server generates a new encryption key which is used to re-encrypt the data 59 so that it can be safely stored again on the client device Local Storage 60. The new encryption key is then stored with the user account in the anticipation of future decryption and use 61.

Example Embodiment

[00080]   Figures 4A, B, C illustrate in diagrammatic form one form of topology and execution of an example embodiment of the invention.

[00081]   In this instance, with reference to figure 4A, a storage system 80 relies on a server 81 in communication with client device 82 in this instance over network 83. In this particular instance network 83 includes the internet which itself comprises an interconnected network of computers adapted for transmission of packet data 84 to destinations identified in header 85. In many circumstances client device 82 will be in electronic communication with network 83 by way of initial link 86 which may comprise a mobile telephone network or other radio communication network such as wifi or

14

the like.

Unsecured Window Period

[00082]   In this instance, client device 82 includes a client
device memory 87 and client devicse processor 88 whereby code
89 stored in memory can be executed by processor 88. In this
instance the code 89 includes an application adapted to
receive commands 90 and data 91 from server 81. In this
instance the commands 90 and data 91 permit creation of form
fields F1 - F7 in a form structure 92. In preferred forms the
form structure 92 is determined on server 81 and forms part of
the commands 90 and data 91 transmitted to client device 82
during the unsecured window period. Each form field can
receive respective data D1 - D7 which may be received from the
server 81 as data 91 or may be inserted by a local user of the
client device 82. The form structure 92 and data 91 is
presented in unsecured form during an unsecured window period
of storage system 80. During this period the data D1 - D7 is
transmitted from client device 82 to server 81. Server 81,
typically located at a location physically remote from client
device 82, received data D1 - D7 over either a secured or
unsecured data channel and saves data D1 - D7 in server memory
93 referenced against an entity account 94.

Secured Window Period

[00083]   With reference to figure 4B a secured window period
commences when server 81 encrypts data D1 - D7 with reference
to at least one key 95 so as to form encrypted data 96 which
is then transmitted by server 81 to client device 82 which is
to say the client device 82 which formulated data D1 - D7 and
from which the data was transmitted to server 81 referenced
against entity account 94. In one form an entity account may

be owned by the client device 82 itself. In another form the
entity account may be owned by a user of the code 89 executed
on the client device 82 from which the data D1 - D7
originated.

[00084]   In either case the server 81 references the data
both before encryption and after encryption against the entity
account 94.

[00085]   In a preferred form the server 81 is a web server.
In an alternative preferred form the server may serve
application functionality in the form of an API or the like
for execution as code 89 on client device 82.

[00086]   Once the data D1 - D7 has been encrypted and
transmitted to client device 82 the server 81 deletes all
instances of data D1 - D7 from its storage - whether in an
encrypted or unencrypted form leaving the only key 95 stored
against the entity account 94.

[00087]   During this secured window period the client device
82 stores the encrypted data 96 locally in client device
memory 87. It is to be noted that the key 95 is not provided
to the client device 82.

Subsequent Use of Data D1 - D7

[00088]   With reference to figure 4C in sessions comprising
execution of code 89 as subsequent sessions to an initial
session on client device 82 an unsecured window session is
initiated during a subsequent unsecured window period during
which execution of code 89 causes transmission of encrypted
data 96 to server 81 where upon, following authentication of
entity account 94 corresponding key 95 is utilised to decrypt

data D1 – D7 and transmit it over either a secured or
unsecured channel from server 81 to client device 82. In a
preferred form the data D1 – D7 will repopulate the form
structure 92 on the client device 82. This data may then be
used for various functionality on the client device 82. The
data D1 – D7 may be amended during the subsequent session
during the subsequent unsecured window period. When a user of
the client device 82 or the client device itself terminates
the subsequent unsecured window period the data D1 – D7 is
transmitted to server 81 as described with reference to figure
4A and a subsequent secured window period is initiated as
described with reference to figure 4B.

[00089]   In a preferred form a user of client device 82 may
be given the option as to whether to invoke the procedure
described above.

Client Server Environment

[00090]   With reference to figure 5 there is illustrated in
block diagram form the topology of a typical client/server
environment 70 which, in this instance, comprises at least one
client device 71 and at least one server 72.

[00091]   The client device 71 is in data communication with
the server 72 via a data communication channel 73. In some
instances this channel 73 may be unidirectional. In other
instances the channel 73 will be bidirectional.

[00092]   In preferred forms the channel 73 or at least part
of it is implemented over the Internet whereby data is
communicated in packet form, each packet comprising at least a
header containing target address information and a data
portion containing a data "payload".

[00093]   In preferred forms data for of a webpage on the
client 71 resides on server 72 and is delivered from the
server 72 to the client 71 on request made by the client to
the server. In some forms the data comprises a series of
commands which, upon execution on the client 71, result in
communication of data on the client device 71-usually by way

of a visual interface, often display of a webpage 74.

[00094]   The webpage 74 itself typically is comprised of a
multitude of data portions-some of which are constructed by
the client 71 executing the commands received from the web
server 72 whilst other portions may be comprised of data
entered into the client by way of local input output operated
by a user of the client device 71.

[00095]   A common command language in use today is the HTML
language.

In Use - EXAMPLE 1

[00096]   Figures 6A-P disclose in block diagram format an
example application of the system.

[00097]   **Fig 6A-What happens initially**

[00098]   With reference to fig 6A an application 101 runs on
the user device 100-in this instance a smart phone.

[00099]   **Fig 6 B-enters form, gets transferred to server**

[000100]   With reference to figure 6B user 102 fills out, in
this example, their flight details and payment details in
fields 105 on a form 103 displayed on touch enabled display
104 of the user device 100.

[000101]   In this instance there are seven fields F 1, F 2… F
7 each containing respective data entries D 1, D 2… D 7.

[000102]   In this instance the user device 100 is in first
data communication with a Web server 106 whereby data
pertaining to a given transaction 108 initiated on user device
100 is exchanged as between the user device 100 and the server
106.

[000103] In some forms the server 106 may itself be in separate second data communication with server 107. Server 107 may for example utilised by financial intermediaries for the purpose of facilitating funds transfer including, for example, payment authorisation as between the parties involved in the transaction 108.

[000104] **Fig 6C-data copied to clearinghouse form (server 107) without storing it/PCI compliant.**

[000105] With reference to figure 6C in this instance user 102 enters data D 12 data D 7 in the fields F1 to F7 constituting the data necessary to trigger transaction 108 whereby the data is communicated electronically to server 106. As necessary server 106 communicates with server 107 for the purpose of payment authorisation of the transaction 108.

[000106] **Fig 6D and fig 6E-first time a user uses your site with application 101 ("wallet") enabled it appears the same except**

[000107] With reference to figures 6D and 6E, during this process the user is issued with an invitation "would you like to securely store your credit card to make future transactions?" The same invitation may be made in respect of any other data which the user wishes to have treated securely-for example other forms of payment data including financial data, medical data and the like.

[000108] **Fig 6F to fig 6P-key generation and use sequence**

[000109] With reference to Fig 6F-an encryption key is generated and the data is encrypted.

[000110]   With reference to fig 6G-the encryption key is stored with the user account.

[000111]   With reference to fig 6H and the encrypted card data is stored back on the user's device and protected by the keychain.

[000112]   With reference to 6I and server memory is dumped so that only the key is on the server and the secure data is on the phone.

[000113]   With reference to 6J on subsequent connections the user is authenticated.

[000114]   And with reference to Fig 6K the encrypted data passed through to the server.

[000115]   With reference to 6L then the key is retrieved from the user's account and the data decrypted and delivered to the clearinghouse.

[000116]   With reference to 6M then the card data is re-encrypted with a new key and

[000117]   With reference to 6N the new key is stored with the user's account and

[000118]   With reference to 6O the new encrypted data chunk is stored back on the user's device for the next transaction.

[000119]   With reference to 6P and memory is cleared leaving only the key on the server and the encrypted data on the user's device.

Alternative Embodiments

[000120]   The example embodiment re-encrypts the data with a
new key every time the data is used. An alternative embodiment
could see the data re-encrypted every time or at a multiple of
times that the client and server connect irrespective of
whether the data is processed and or used in a form or not.
Conversely another embodiment could see the encrypted data
using the same encryption key for an unlimited amount of time.

[000121]   The example embodiment uses HTML5 Local Storage to
store the encrypted data on the client. An alternative
embodiment could use any storage means available on the client
and accessible to a server on a web enabled application or
browser.

[000122]   The example embodiment is used in the context of
data stored and linked to a single domain. An alternative
embodiment could allow the encrypted data to be accessed
across multiple domains using a technique known in the art. An
example of this is where an iFrame allowing access to local
storage for a secure domain area where encrypted data is
stored, can be embedded within a web page that accesses and
references a different secure domain space on the client, but
where data from the iFrame secure area can be passed to and
used by the parent or other related pages.

[000123]   The example embodiment shows the user being able to
optionally encrypt and store the data for future use. An
alternative embodiment could see this function being an
automatic process without the need for the user to opt for the
data to be encrypted and stored locally.

CLAIMS

1. In a client server environment a method of securely
   storing data; said method comprising
   generating a data element at a second location;
   transmitting the data element to a first location
   separate and remote from the second location;
   encrypting the data element at the first location thereby
   to form an encrypted data element;
   transmitting the encrypted data element to the second
   location separate and remote from the first location and
   storing the encrypted data element at the second
   location; and wherein the second location is constituted
   as a client device.

2. The system of claim 1 wherein a decryption key for the
   encrypted data element is generated at the first location
   and stored at the first location or within its network
   environment.

3. The system of claim 1 or 2 wherein the first location is
   constituted by a server device and the key is stored on
   the server device or within its network environment.

4. The system of claim 1 or 2 or 3 wherein the client device
   is programmed to execute a Web enabled application which
   has a storage capability.

5. The system of any one of claims 1 to 4 wherein the
   encrypted data element is stored on the client device
   utilising the storage capability of the Web enabled
   application.

6. The system of claim 4 claim 5 wherein the Web enabled application is a Web browser.

7. The system of claim 6 wherein the web browser executes HTML 5 including the HTML 5 local storage function.

8. In a client server environment a method for storing data securely; said method comprising encrypting the data at a first location using a processor located at the first location; the data thus encrypted comprising encrypted data which requires a key to decrypt the encrypted data; the key stored at the first location or within its network environment; the encrypted data transmitted to a second location remote from the first location and stored at the second location until further processing is required to be done on the data at which time the encrypted data is transmitted to the first location and unencrypted by applying the key to a decryption algorithm executed by the processor.

9. In a client server environment an apparatus for secure storage of data; said apparatus comprising a first processor at a first location which encrypts data utilising a key; said apparatus further comprising a second processor located at a second location remote from the first location; the data, after encryption, moved over a network to the second processor and stored in association with the second processor until the data is required for execution of an application on the first processor at which time the data is moved back from the second processor to the first processor and the first processor applies the key to a decryption algorithm to decrypt the data for use by the application executing on

the first processor; the data accessible at the second
location during an unsecured window period.

10. A method of isolation from its key of data encrypted by
    means of the key and whereby separate retrieval of the
    key by a user for decryption of the data is not
    required; said method comprising, in a client/server
    environment, receiving data from the server during an
    unencrypted window period; working on the data on the
    client during the unencrypted window period;
    transmitting the data to the server at the end of the
    unencrypted window period; the server encrypting the
    data with reference to the key to form encrypted data;
    the server then transmitting the encrypted data to the
    client; the server then deleting all of the data and the
    encrypted data on the server thereby terminating the
    unencrypted window period; the key remaining stored only
    on the server.

11. A method of isolation from its decryption key of
    encrypted data stored on a client device; said method
    comprising
    storing the decryption key referenced against a username
    account/ user login-data for the client device on a
    separate device;
    retrieving the data by transferring the encrypted data
    to the separate device and upon authentication of the
    user login/username account decrypting the encrypted
    data using the decryption key.

12. The method of claim 10 or 11 wherein the separate device
    is a Web server.

13. The method of claim 10 or 11 or 12 wherein the user login is a username and password.

14. The method of claim 10 or 11 or 12 or 13 wherein in a client server environment during a secured non-use window data is stored securely on the client as encrypted data requiring decryption by decrypting the encrypted data and wherein the decryption key is not stored on the client.

15. The method of any one of claims 10 to 14 wherein the decryption key is stored on the server.

16. The method of any one of claims 10 to 15 wherein the encryption/decryption is not done on the client.

17. The method of any one of claims 10 to 16 wherein the encryption/decryption is done on the server.

18. The method of any one of claims 10 to 17 wherein the decryption key is released for use if and only if authentication of a client/service session is validated.

19. The method of any one of claims 10 to 18 wherein the environment requires authentication at the user level.

20. The method of any one of claims 10 to 19 wherein decryption key is referenced against the authentication.

21. The method of any one of claims 10 to 20 wherein authentication is at the user device level.

22. The method of any one of claims 10 to 21 wherein the data is a small data quantum.

23. The method of any one of claims 10 to 22 wherein the data is only changed incrementally, if at all, each user session.

24. The method of claim 22 or claim 23 wherein the data quantum is 100 MB.

25. The method of claim 22 or claim 23 wherein the data quantum is 50 MB.

26. The method of claim 22 or claim 23 wherein the data quantum is 10 MB.

27. The method of claim 22 or claim 23 wherein the data quantum is 1 MB.

28. The method of claim 22 or claim 23 wherein the data quantum is 0.5 MB.

29. The method of claim 22 or claim 23 wherein the data quantum is 0.1 MB.

30. The method of claim 22 or claim 23 wherein the data quantum is separated into sub portions for transmission from the second location to the first location.

31. The method of claim 22 or claim 23 wherein the data quantum is separated into sub portions for transmission from the first location to the second location.

32. The method of claim 31 wherein each sub portion is less than 4 MB.

33. The method of any one of claims 10 to 32 implemented as
    instructions stored on a medium; the instructions when
    executed implementing the method.

34. Apparatus incorporating a processor in communication
    with a medium; the medium having stored thereon
    instructions to give effect to the method of claim 33.

35. A server which implements the method of any one of
    claims 10 to 33.
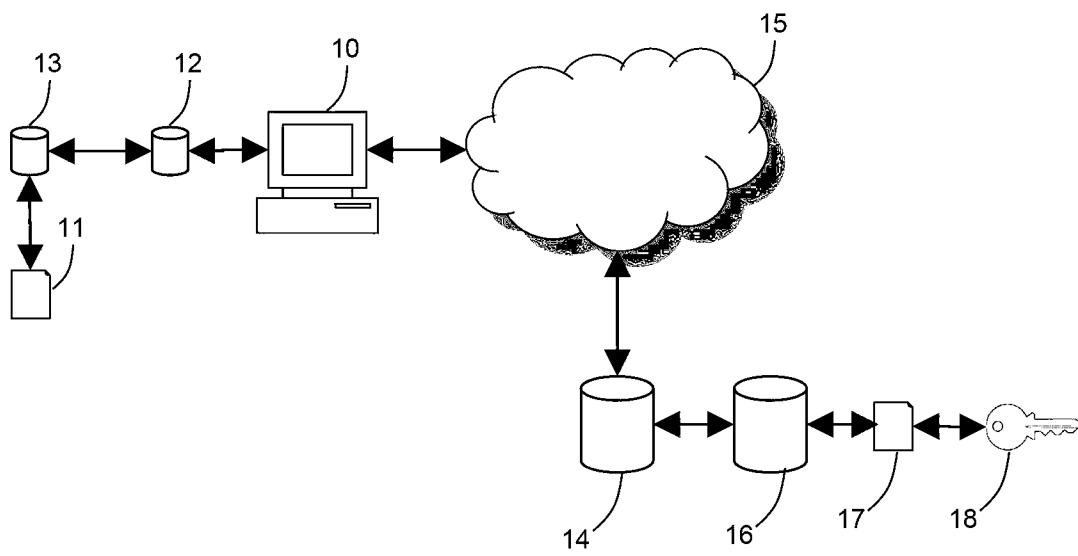
36. A client device which implements the method of any one
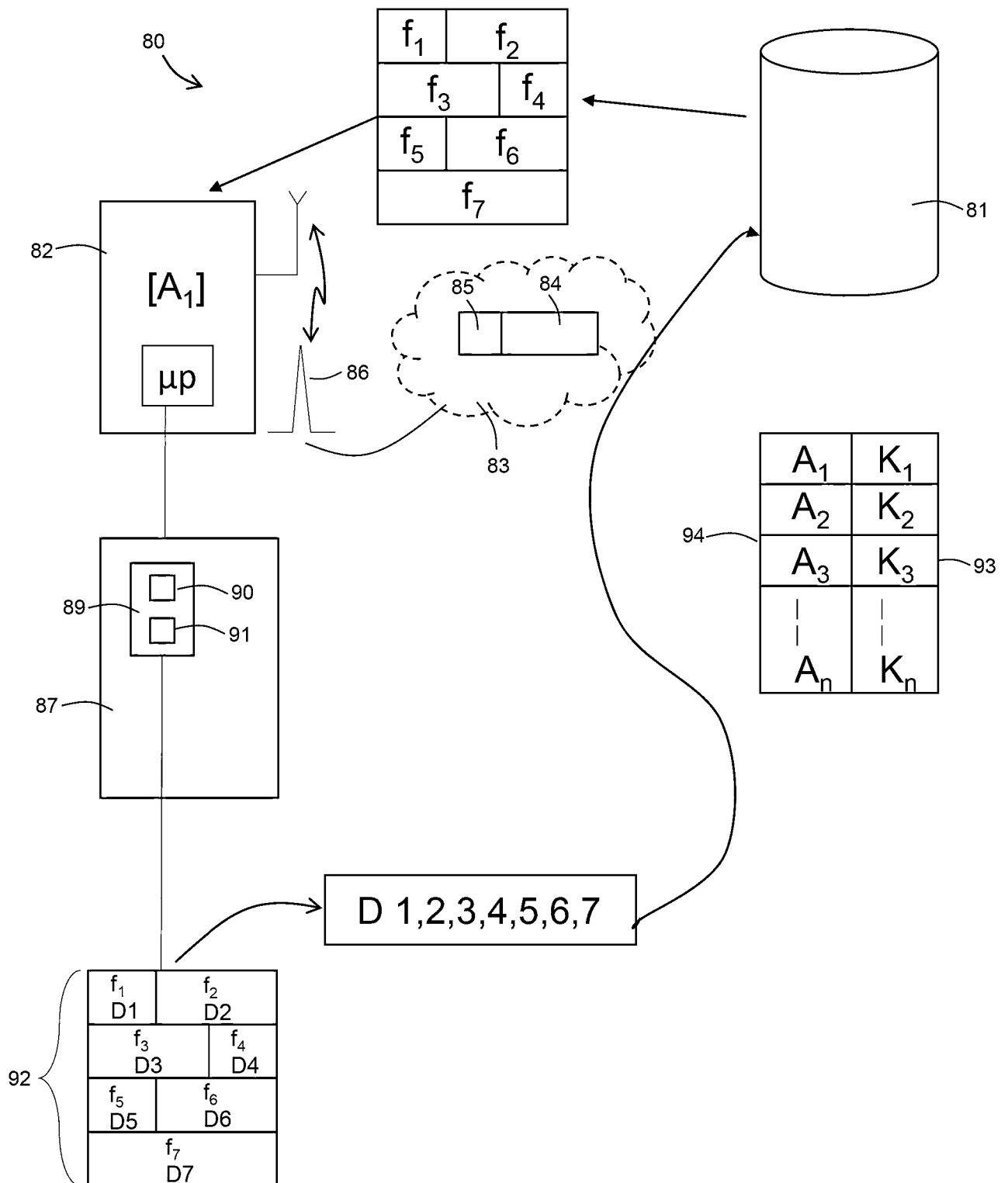   of claims 10 to 33.

1/23

Figure 1

2/23

Figure 2

User initiates secure connection — 50

User client logged onto secure user account — 51

Server supplies form for formatting of data to be collected and stored with option to use stored data on client — 52

User opts to use client side encrypted stored data in form — 53

Encrypted Data is retrieved by the server — 54

Encryption key is retrieved by the server from the user account — 55

Data is decrypted — 56

Data is processed — 57

Server generates new encryption key — 58

Encrypted data is stored on the client in Local Storage for future retrieval — 60

Data is encrypted — 59

New key is linked to the user account — 61

62

63

Figure 3

Unsecured Window Period

Fig. 4A

5/23

82

[A$_1$]

µp

Encrypted Data

D1 – D7

81

94

A$_1$	K$_1$
A$_2$	K$_2$
A$_3$	K$_3$
A$_n$	K$_n$

95

Secured Window Period

Fig. 4B

Subsequent Unsecured Window Period

Fig. 4C

Second Location   First Location

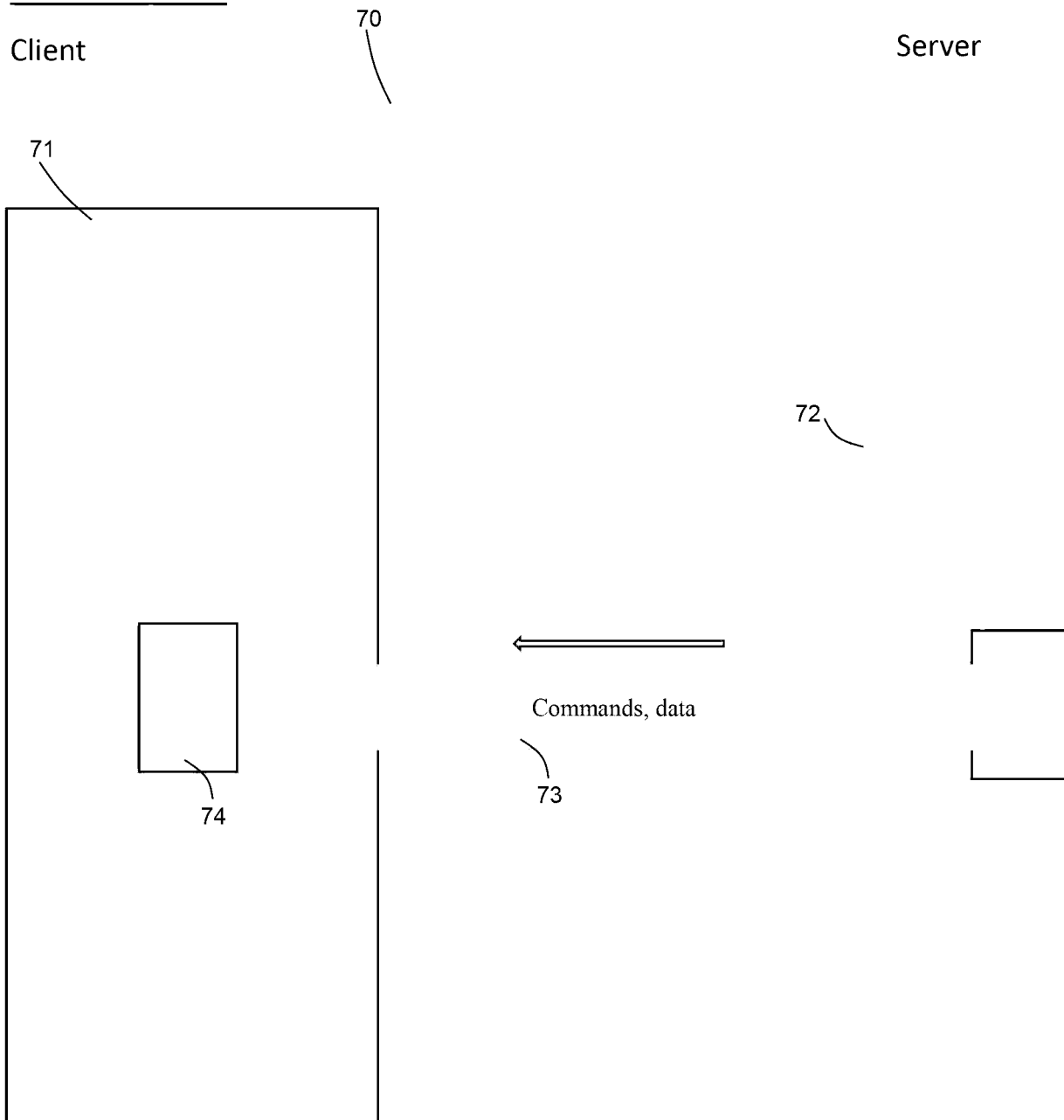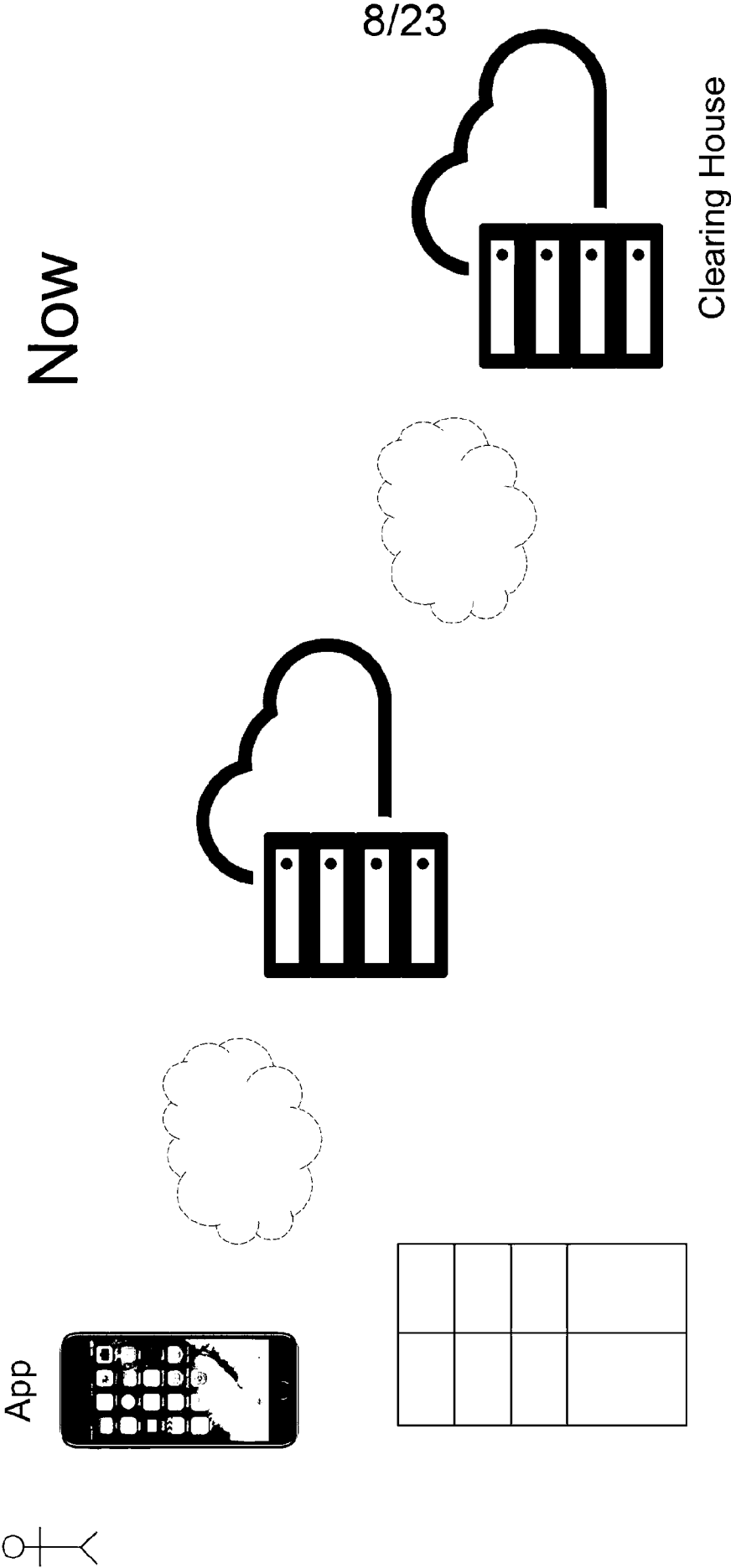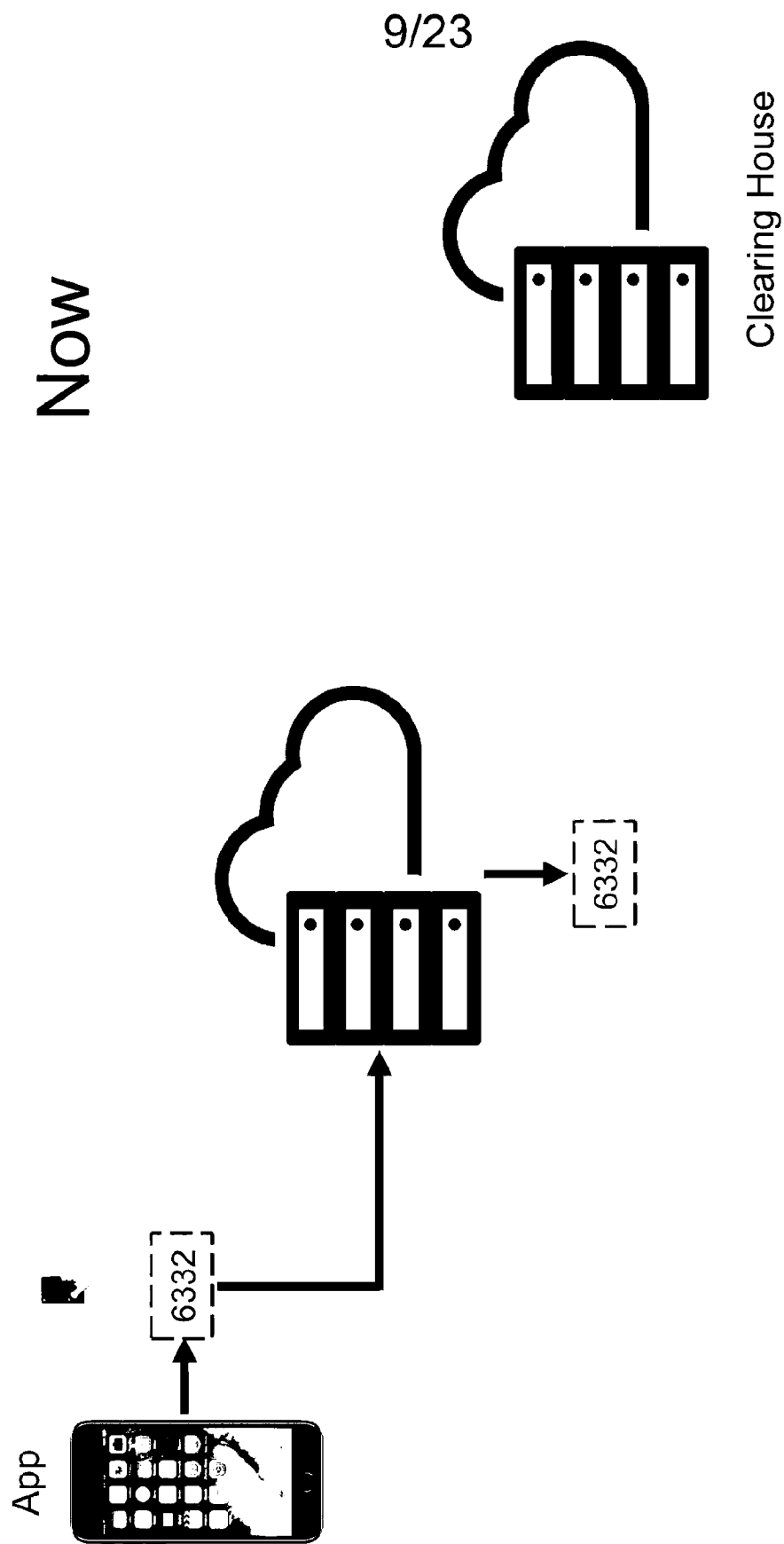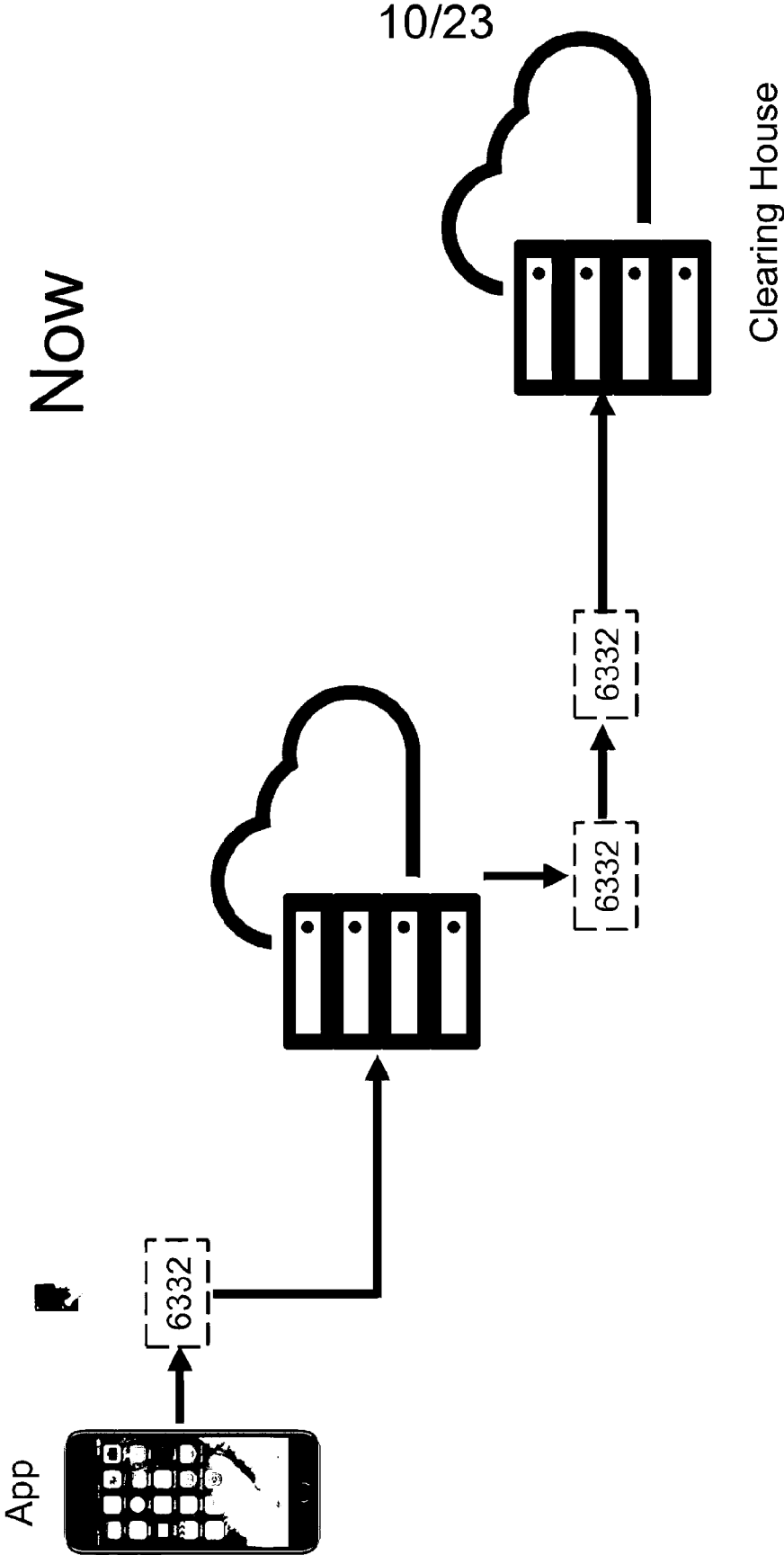Client   Server

70

71

72

Commands, data

73

74

Fig. 5

Now

Clearing House

App

Fig. 6A

Now

Clearing House

6332

App

6332

Fig. 6B

10/23

Now

Clearing House

6332

6332

6332

App

Fig. 6C

11/23

1st Time

App

6332

6332

6332

Clearing House

Fig. 6D

1st Time

12/23

Clearing House

Would you like to
securely store your
credit card to make
future transactions?

6332

XXXX

App

Fig. 6E

13/23

1st Time

Clearing House

6332

6332

XXXX

App

Fig. 6F

14/23

1st Time

Username

Clearing House

6332

6332

XXXX

App

Fig. 6G

15/23

1st Time

Fig. 6H

16/23

1st Time

Clearing House

Username

App

iPhone Keychain

XXXX

Fig. 6I

Subsequent

Clearing House

App

iPhone
Keychain

XXXX

Fig. 6J

Subsequent

Clearing House

App

iPhone
Keychain

Fig. 6K

Fig. 6L

Subsequent

Clearing House

Username

6332

XXXX → YYYY

App

iPhone Keychain

XXXX

Fig. 6M

21/23

Subsequent

Clearing House

Username

App

iPhone
Keychain

Fig. 6N

Subsequent

Clearing House

Username

App

iPhone
Keychain

Fig. 6O

Subsequent

Username

Clearing House

App

iPhone
Keychain

[YYYY]

Fig. 6P

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/62 (2013.01)   H04L 9/08 (2006.01)

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

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

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

EPODOC, WPIAP and TXTE: CPC Marks-G06F21/60/low and H04L9/08. Keywords-encrypt, decrypt, secure, send, transmit, receive, transfer, return, store, data, file, information, location, distributed, remote, server side, computer, server, module, interface, database, first, second, web, internet, browser, page, site, application, username, account, login, details or similar terms.

ESPACENET and Google Patents: G06F21/62, H04L9/08, storage, secure, data, client side, local, remote, key, transfer, encrypted, server, client, location, first, second, generate, return or like terms.

Applicant/Inventor Search Databases: ESPACENET and IP Australia Internal Databases (PAMS NOSE and INTESS).

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	"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
"E"	earlier application or patent but published on or after the international filing date	"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
"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)	"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
"O"	document referring to an oral disclosure, use, exhibition or other means	"&"	document member of the same patent family
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search	Date of mailing of the international search report
28 March 2017	28 March 2017

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	Pearce Van De Pol AUSTRALIAN PATENT OFFICE (ISO 9001 Quality Certified Service) Telephone No. 02 6222 3629

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2011/0307383 A1 (RATICA) 15 December 2011 Paragraphs 0041, 0049, 0051, 0053-0054, 0056, 0060, 0065 and 0066-0067, and Figure 1.	1-36
A	WO 2013/020178 A1 (COCOON DATA HOLDINGS LIMITED) 14 February 2013 Whole Document	
A	WO 2013/020177 A1 (COCOON DATA HOLDINGS LIMITED) 14 February 2013 Whole Document	
A	US 2007/0198823 A1 (BLEW et al.) 23 August 2007 Whole Document	
A	US 2015/0006890 A1 (AMAZON TECHNOLOGIES, INC) 01 January 2015 Whole Document	

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**
US 2011/0307383 A1	15 December 2011	US 2011307383 A1	15 Dec 2011
		AU 2011264813 A1	24 Jan 2013
		AU 2011264813 B2	30 Jul 2015
		AU 2015249057 A1	12 Nov 2015
		CA 2802071 A1	15 Dec 2011
		EP 2580724 A1	17 Apr 2013
		JP 2013529804 A	22 Jul 2013
		JP 6084564 B2	22 Feb 2017
		WO 2011156568 A1	15 Dec 2011
		ZA 201209349 B	25 Sep 2013
WO 2013/020178 A1	14 February 2013	WO 2013020178 A1	14 Feb 2013
WO 2013/020177 A1	14 February 2013	WO 2013020177 A1	14 Feb 2013
US 2007/0198823 A1	23 August 2007	US 2007198823 A1	23 Aug 2007
		US 7519810 B2	14 Apr 2009
		US 2004181668 A1	16 Sep 2004
US 2015/0006890 A1	01 January 2015	US 2015006890 A1	01 Jan 2015
		US 9286491 B2	15 Mar 2016
		AU 2014216607 A1	06 Aug 2015
		CA 2898995 A1	21 Aug 2014
		CA 2899008 A1	21 Aug 2014
		CA 2899014 A1	21 Aug 2014
		CA 2899019 A1	21 Aug 2014
		CA 2899027 A1	21 Aug 2014
		CA 2916954 A1	08 Jan 2015
		CN 105027130 A	04 Nov 2015
		CN 105103119 A	25 Nov 2015
		CN 105103488 A	25 Nov 2015
		CN 105122265 A	02 Dec 2015
		CN 105191207 A	23 Dec 2015
		CN 105493435 A	13 Apr 2016
		EP 2956852 A1	23 Dec 2015
		EP 2956880 A1	23 Dec 2015
		EP 2956888 A1	23 Dec 2015

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

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**
		EP 2957063 A1	23 Dec 2015
		EP 2957065 A1	23 Dec 2015
		EP 3017561 A1	11 May 2016
		JP 2016508643 A	22 Mar 2016
		JP 2016508699 A	22 Mar 2016
		JP 2016511994 A	21 Apr 2016
		JP 2016511995 A	21 Apr 2016
		JP 2016515235 A	26 May 2016
		JP 2016524429 A	12 Aug 2016
		KR 20150119112 A	23 Oct 2015
		SG 11201505742P A	28 Aug 2015
		SG 11201510650U A	28 Jan 2016
		US 9300464 B1	29 Mar 2016
		US 9300639 B1	29 Mar 2016
		US 9367697 B1	14 Jun 2016
		US 2014230007 A1	14 Aug 2014
		US 9547771 B2	17 Jan 2017
		US 2014229729 A1	14 Aug 2014
		US 9590959 B2	07 Mar 2017
		US 2014229732 A1	14 Aug 2014
		US 2014229737 A1	14 Aug 2014
		US 2014229739 A1	14 Aug 2014
		US 2015019858 A1	15 Jan 2015
		US 2016191237 A1	30 Jun 2016
		US 2016196438 A1	07 Jul 2016
		US 2016283723 A1	29 Sep 2016
		WO 2014126813 A1	21 Aug 2014
		WO 2014126814 A1	21 Aug 2014
		WO 2014126815 A1	21 Aug 2014
		WO 2014126816 A1	21 Aug 2014
		WO 2014126882 A1	21 Aug 2014
		WO 2015002875 A1	08 Jan 2015

End of Annex

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

Form PCT/ISA/210 (Family Annex)(July 2009)