(51) International Patent Classification:
H04L 29/06 (2006.01)  H04L 12/22 (2006.01)

(21) International Application Number:
PCT/EP2009/061894

(22) International Filing Date:
15 September 2009 (15.09.2009)

(25) Filing Language:
English

(26) Publication Language:
English

(71) Applicant (for all designated States except US): NOKIA SIEMENS NETWORKS OY [FI/FI]; Karaportti 3, FI-02610 Espoo (FI).

(72) Inventors and
(75) Inventors/Applicants (for US only): CHEN, You, Lei [CN/CN]; No. 1405, Building 1 BalizhuangDongLi, Chaoyang District, Beijing. 100025 (CN). LIU, Jin [CN/CN]; Room A706, Zhong Xin Jia Yuan, Mananyuan, BeiJi Jia Yi Hao (No.1A), Haidian, District, Beijing, 100191 (CN). SUN, Shao, Jun [CN/CN];

(54) Title: METHODS AND SYSTEMS FOR DELEGATING AUTHORIZATION

(57) Abstract: A delegation protocol is described, which enables an application to obtain data, such as user credentials, from a service provider. The delegation protocol is controlled by a permit that is issued by the service provider. The permit includes information such as the username of the user at both the application and the service provider, the identity of the application and detailed authorisation which defines the information that can be shared with the application. The permit is signed by a key based on the user's password at the service provider.
TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published: — with international search report (Art. 21(3))
METHODS AND SYSTEMS FOR DELEAGATING AUTHORIZATION

The invention is directed to identity management and service access control.

A “mash-up” service or application refers to the combination of disjoint services and/or applications to share information and provide unified services. An example of a mashup is the use of cartographic data from a mapping application to add location information to real estate data, thereby creating a new and distinct service that was not originally provided by either source. An advantage of such mash-up services is that new resources can be easily created since the mash-up services use or combine existing resources.

“Delegation” involves the technology of assignment of authority and responsibility to another entity to carry out specific activities. In a mashup system, for example, delegation solves the problem of how to authorise a mashup application to obtain resources from other services on behalf of the user, and also guarantee that the user’s right to the resource is not abused.

By way of example, when accessing an Internet service, user data (such as personal attributes, log, pictures, video data etc.) are typically stored at the service provider. If third parties want to access the user information, one way is for each third party to sign an agreement with the service provider. Such agreements usually specify the information that can be shared and the corresponding rights and obligations. However, in most cases, service providers and third parties don’t have such an agreement; delegated authorization was developed to address this problem. Delegated authorization enables third parties to get authorization from a user and to access user information on behalf of the user.
Figure 1 is a block diagram of a system, indicated generally by the reference numeral 1, comprising a user 2, an application 4 and a service provider 6. The application 4 may be a mash-up application. The application may be an identity management system (IDM), which can be thought of as a mash-up application.

The service provider 6 stores data concerning the user 2 and the user wishes to provide the application 4 with authority to access the data stored at the service provider directly. The simplest method by which the user might delegate authority to the application 4 is simply to inform the application 4 of the required user credentials (such as the user’s username and password pair) at the service provider. Of course, this method requires the user 2 to have complete trust in the application 4, which may not always be the case. Indeed, this method violates a general security principle that credentials (such as passwords) should only be presented to the party that has issued the credentials. Moreover, such an arrangement does not enable the service provider 6 to distinguish between requests from the user 2 and the application 4.

In the real world, most users only want to provide delegated authorization to non-sensitive data stored at a service provider. For example, if the application 4 is an online printing application and the user 2 has digital images stored at the service provider 6, the user might allow the online printing application to read the images stored at a service provider, but that user might not share sensitive data such as bank account and credit card details with mashup applications other than the bank itself.

The present invention seeks to address at least some of the problems outlined above.
In accordance with an aspect of the invention, there is provided a method comprising: receiving (for example, at a service provider) a delegation authorisation request (typically received from an application) identifying a resource (such as an identity provider or an identity management system or some other application, or some form of data) for which delegated authorisation is requested (such as an application wanting delegated authorisation); determining (at the service provider) whether or not the delegation authorisation request is authorised by a user; and creating a delegation permit, the permit defining information for which delegated authorisation is granted.

In accordance with an aspect of the invention, there is provided an apparatus (for example at a service provider or forming part of a service provider) comprising: a first input for receiving a delegation authorisation request identifying a resource (such as an application) for which delegated authorisation is requested (e.g. an application wanting delegated authorisation); a first processor for determining whether or not the delegation authorisation request is authorised by a user; a second processor (which may be the same as the first processor) adapted to create a delegation permit, wherein the delegation permit includes one or more of a username of the user at a service provider including said resource, a policy identifying one or more resources for which delegated authorisation is granted (e.g. a resources or some other data stored at the service provider that can be shared with the application) and a digital signature (for example, based on a user password); and a first output for outputting the delegation permit (to the application, possibly by redirection).

The invention typically includes forwarding the delegation permit to the entity (typically an application) being granted delegated authorisation. The permit is often forwarded via the user, using redirection, although the permit could be sent directly to the entity concerned.
The permit may include one or more (typically all) of a username of the user at a service provider including said resource, a policy identifying one or more resources for which delegated authorisation is granted and a digital signature (for example, based on a user password). The permit may alternatively, or additionally, include one or more of: a username for the user at an application for which delegated authorisation is granted; an identity for the application; an expiration time for the permit; an issue time for the permit; and a local identity of the service provider itself.

The delegation permit may be signed using a key based on a value specific to the user (e.g. a password of the user at the service provider). It should be noted that the security of the permit does not need to exceed the security of the password, since if a third party has access to the user’s password, then access to the service can be obtained without need to use the algorithms of the present invention.

In order to determine whether or not the delegation authorisation request is authorised by a user, a prompt may be presented to the user. For example, an apparatus in accordance with the invention may include a second output for presenting a prompt to the user requesting an indication of whether the delegation authorisation request is authorised by the user and a second input for receiving a response to the prompt, wherein the first processor determines whether or not the delegation authorisation request is authorised by a user on the basis of said response to said prompt.

The delegation authorisation request may be received from an application requesting delegated authorisation. Alternatively, the delegation authorisation request may be received from the user (the user thereby granting delegated authorisation to the application). Further, determining whether or not the delegation authorisation request is
authorised by a user may include determining that the request is received from the user.

The invention may further comprise: receiving, at a service provider, an access request (e.g. in the form of a request for delegated access to a resource) from the application (wherein the access request typically requests access to a resource), wherein the request includes the said permit; and checking (typically at the service provider) the details of the permit to determine whether the access request is authorised (e.g. whether delegated authorisation to access a requested resources has been granted). For example, an apparatus in accordance with an aspect of the invention may include: a third input for receiving a request for access to one or more resources from the application, wherein the request includes the said permit; and a third processor for checking the details of the permit to determine whether the access request is authorised.

In accordance with an aspect of the present invention, there is provided a method comprising: receiving at a service provider, an access request (e.g. in the form of a request for delegated access to a resource) from an application (such as an IDP or an IDM or some other application), wherein the request includes a permit; checking (at the service provider) the details of the permit to determine whether access to the service is authorised (e.g. whether delegated authorisation to access a requested resource has been granted); and providing access to the service in the event that access is deemed to be authorised.

In accordance with an aspect of the present invention, there is provided an apparatus comprising: a first input for receiving an access request (e.g. a delegated access request) from an application (such as an IDP or an IDM), wherein the request includes a permit; a first processor adapted to check the details of the permit to determine whether access to the service is authorised (e.g. whether delegated access has been
authorised); and a second processor adapted to provide access
to the service in the event that access is deemed to be
authorised. The first processor may be adapted to: extract
an identity for the user from the permit; and use the user
identity to determine a key (possibly based on the user’s
password) used to sign the permit from the user in order to
verify the permit signature.

The permit typically includes one or more of: a username of
the user at a service provider including said resource; a
policy identifying resources stored at the service provider
for which delegated authorisation is granted (e.g. that can
be shared with the application); a digital signature (for
example, based on a user password); a username for the user
at an application; an identity for the application; an
expiration time for the permit; an issue time for the permit;
and a local identity of the service provider itself.

In some forms of the invention, checking the details of the
permit to determine whether the access request is authorised
comprises: extracting an identity for the user from the
permit; and using the user identity to determine a key
(possibly based on the user’s password) used to sign the
permit from the user in order to verify the permit signature.
Checking details of the permit may also include checking an
expiry time of the permit to determine that the permit
remains valid.

In accordance with a further embodiment of the invention,
there is provided a delegation authorisation permit, the
permit comprising: a username for a user at a service
provider; a policy identifying resources stored at the
service provider for which delegated authorisation is granted
(e.g. that can be shared with the application); and a digital
signature, wherein the signature is calculated using a key
generated from a value specific to the user (such as a
password). The permit may alternatively, or additionally,
include one or more of: a username for the user at an
application; an identity for the application; an expiration time for the permit; an issue time for the permit; and a local identity of the service provider itself.

In forms of the invention including more that one input and/or more than one output, the various inputs and/or outputs may be provided using at least some of the same resources. For example, a first input and a second input may be provided using a single physical input. Similarly, a first input and a first output may be provided using a single physical input-output pin. In a similar way, in forms of the invention including more than one processor, the processor functionality may be provided by a single processor. For example, the functionality of a first, second and third processor may be provided by a single physical processor, or by two physical processors, or by three physical processors. Furthermore, the functionality of a particular processor of the invention may be distributed over multiple processors.

According to an aspect of the invention, there is provided a computer program comprising: code (or some other means) for receiving (for example, at a service provider) a delegation authorisation request identifying a resource (such as an identity provider or an identity management system or some other application) for which delegated authorisation is requested (such as an application wanting delegated authorisation); code (or some other means) for determining (typically at the service provider) whether or not the delegation authorisation request is authorised by a user; and code (or some other means) for creating a delegation permit, the permit defining information for which delegated authorisation is granted (e.g. that can be shared with an application requesting delegated access). The computer program may be a computer program product comprising a computer-readable medium bearing computer program code embodied therein for use with a computer.
According to an aspect of the invention, there is provided a computer program comprising: code (or some other means) for receiving at a service provider, an access request from an application (such as an IDP or an IDM or some other application), wherein the request includes a permit and wherein the request requests access to a resource; code (or some other means) for checking (typically at the service provider) the details of the permit to determine whether access to the resource is authorised; and code (or some other means) for providing access to the resource in the event that access is deemed to be authorised. The computer program may be a computer program product comprising a computer-readable medium bearing computer program code embodied therein for use with a computer.

Exemplary embodiments of the invention are described below, by way of example only, with reference to the following schematic drawings.

Figure 1 is a block diagram of a system for enabling service access delegation;
Figure 2 is a block diagram showing an algorithm in accordance with an aspect of the present invention;
Figure 3 is a block diagram showing an algorithm in accordance with an aspect of the present invention;
Figure 4 is a block diagram showing an algorithm in accordance with an aspect of the present invention;
Figure 5 is a message sequence in accordance with an aspect of the present invention; and
Figure 6 is a message sequence in accordance with an aspect of the present invention.

The present invention provides a protocol that seeks to address a number of the problems outlined above. The protocol introduces a permit that defines everything needed for a data retrieval operation. The permit is issued by the service provider 6, often in negotiation with the user, but not necessarily since the permit may be predefined without
negotiation with the user. The permit is provided to the
application 4. Once the application is in possession of a
valid permit, the application can request access to user data
stored at the service provider 6 and, provided the request is
in accordance with the permit, the data is provided to the
application without involving the user. Thus, delegated
access to user data is enabled and controlled by the use of
the permit.

10 The permit includes at least some of the following
information:

• The user’s username at the service provider 6
• The user’s username at the application 4
• The identity of the application 4
• Detailed authorization information which defines which
data can be shared with the application 4
• A location identity of the service provider 6
• An expiration time for the permit
• An issue time for the permit
• Signature of the permit

By way of example, an exemplary permit may take the
following form:

uid=jerry01:site=sp.com:attributelist=credit+age:signature=E7AHv9be7f4Tnc5d+C7CFQ==

Where:
"jerry01" is the user’s username at the service
provider;
"sp.com" is the address of the service provider;
"credit" and "age" refer to the resources for which
delegated access has been granted; and
"E7AHv9be7f4Tnc5d+C7CFQ" is the permit’s signature.

35 In some embodiments of the invention, the inclusion of the
user’s username at the service provider 6, detailed
authorization information which defines which data can be shared with the application 4, and the signature are essential, but each of the other fields is optional. Of course, many alternative options exist. It should be noted that the user’s username at the service provider and the user’s username at the application could be of any form that enables the service provider and the application to locally identify the user.

The permit fields may be signed using a key generated from the user's password at the service provider 6. The use of a signed permit not only ensures that the application 4 cannot forge a permit which is not authorized by the user 2, but also enables the service provider 6 to know that the delegation request was granted by the user. Since key negotiation is not needed, the protocol is kept simple.

Figure 2 is a block diagram of an algorithm, indicated generally by the reference numeral 10, showing an exemplary use of the permit described above.

The algorithm 10 starts at step 12, where a delegation access request is sent from the application 4 to the service provider 6. The delegation access request includes a request for a particular resource (such as a user credential) stored at the service provider and also includes the permit. The algorithm 10 moves to step 14, where the username of the user 2 at the service provider is extracted from the permit.

The algorithm 10 moves to step 16 where, on the basis of the username extracted at step 14, the service provider determines the key for the permit and uses the key to verify the signature in the permit. If the signature is verified, then the algorithm 10 moves to step 18, where the time is checked to determine whether the permit is valid (if an expiration time is provided). Note that order of step 16 and step 18 can be exchanged.
Below is an example of verifying the signature. Suppose the following permit is received:

    uid=jerry01:site=sp.com:attributelist=credit+age:signature=e=E7AHv9be7f4Tnc5d+C7CFQ==

First the service provider splits the whole string with ":" into a list of name and value pairs. Then the service provider parses each of them into name and value and determines that "jerry01" is the user name at the service provider. The service provider then finds the key stored for the user and generates a signature using a pre-defined algorithm such as HMAC-SHA-1 based on the permit string "uid=jerry01:site=sp.com:attributelist=credit+age" (i.e. excluding the signature part of the permit). The generated signature can then be BASE64 encoded and compared to the parsed signature. The verification is considered passed if the two string equal to each other, otherwise the verification fails.

Note the key used to generate or verify the signature is generated and stored when the user’s password is provisioned or changed. The user’s password can be used as a seed to generate the secure key at this time for enhanced security.

If the various steps of the algorithm 10 are passed, then the resource or data requested in step 12 is returned to the application 4.

It should be noted that the permit security does not generally need to exceed the security of the user’s password at the service provider, since if the user’s password is obtained by an attacker, then that attacker can gain access to the user’s details at the service provider without needing to make use of the permit system.

The algorithm 10 described above shows how the permit can be used. Figures 3 and 4 demonstrate how the permit can be
generated. Figure 3 shows an algorithm, indicated generally by the reference numeral 20, in which a permit request is initiated by an application (such as the application 4) wanting access to user data. Figure 4 shows an algorithm, indicated generally by the reference numeral 30, showing an algorithm in which a permit request is initiated by a user (such as the user 2) whose data is stored at the relevant service provider.

The algorithm 20 shown in Figure 3 starts at step 22, where a delegation request is made by an application (such as the application 4) wanting delegated authorization to access a service provider (such as the service provider 6) on behalf of a user (such as the user 2). The delegation request is sent from the application to the service provider. Next, at step 24, the service provider contacts the user to enable the user to confirm that delegated authorization should be granted to the application making the request. If the user confirms that the application should be granted delegation authorization, then the service provider creates a permit and forwards the permit to the application (step 26).

The algorithm 30 shown in Figure 4 starts at step 32, where a user (having logged in at the service provider) defines delegation permissions. Next, at step 34, delegated authorization for a particular application is granted by the user. Finally, at step 36, the service provider generates a permit and forwards the permit to the application concerned.

Thus, the algorithms 20 and 30 both result in delegated authorization being granted to the application, under the control of the user. However, as mentioned above, in the algorithm 20 the process is initiated by the application requesting delegated authorization and in the algorithm 30 the process is initiated by the user granting delegated authorization to the application.
Figure 5 is a message sequence, indicated generally by the reference numeral 40, demonstrating a use of the system 1 in accordance with an aspect of the present invention. Figure 6 is a message sequence, indicated generally by the reference numeral 60, demonstrating an alternative use of the system 1 in accordance with an aspect of the present invention. As described in detail below, the message sequence 40 makes use of the algorithms 10 and 20 described above and the message sequence 60 makes use of the algorithms 10 and 30 described above.

The message sequence 40 shown in Figure 5 starts with the user 2 sending a message 42 to the application 4 to login at the application and to instruct the application to retrieve data from the service provider 6.

In response to the message 42, the application 4 sends a delegation authorization request 44 to the service provider 6 (implementing the step 22 of the algorithm 20). As shown in Figure 5, the delegation authorization request 44 is sent via the user 2 (using redirection). Since the request 44 is sent via the user, it is not necessary for the application 4 to know an address for the service provider 6. This keeps the protocol 40 relatively simple.

The delegation authorization request 44 includes the following information:

- user name for the user 2 at the application 4
- details of the service provider resource or data being requested
- a callback uniform resource locator (URL) for the application 4 where the permit should be sent

An exemplary format of the request message is:

HTTP GET
Where:

"sp.com" is the address of the service provider 6;
"jerry" is the user’s username at the application 4;
"credit" refers to the requested resources; and
www.application.com/callback is the callback URL for the
application 4.

When the service provider 6 receives the delegation request,
the service provider prompts the user 2 to indicate whether
the delegation request should be accepted (implementing step
24 of the algorithm 20). The prompt 46 may take the form of
a web page being presented to the user having the following
form:

<username>@<IDM_URL> is requesting to access your resources
<......>
Do you agree?

If the user agrees to share the requested resource, the
service provider generates a permit, then generates a key
using the user’s password and finally signs the permit with
the key (implementing step 26 of the algorithm 20). The
service provider 6 then transmits the permit to the
application 4, e.g. by encoding the permit as part of the
redirect URL to the user, e.g.
location=http://idm.org/delegate?user=<...>&sp_url=http://api.sp.com/get?permit=<...>.

As shown in Figure 5, the permit may be sent from the service
provider 6 to the application 4 via the user 2, using
redirection (see message 50). The permit can also be sent
directly from the service provider 6 to application 4.

At this stage, the algorithm 20 described above has been
completed and the permit created by the service provider 6
has been passed to the application 4.
The application 4 (such as an IDM) can now request access to details regarding the user stored at the service provider. In order to request such details, the application 4 simply sends a request 52 to the service provider. The request 52 may take the form of a simple HTTP GET request, including the permit created by the service provider 6, sent to the URL of the service provider as provided in the message 50. Thus, the message 52 implements the step 12 of the algorithm 10 described above.

An example for the message 52 can be:
GET
http://www.sp.com/userprofile?user=jerry01&attribute=credit&permit=
uid=jerry01:site=sp.com:attributelist=credit+age:signature=E7AHv9be7f4Tnc5d+C7CFQ==

Where:

    www.sp.com/userprofile?user=jerry01&attribute=credit defines the resource requested; and
    uid=jerry01:site=sp.com:attributelist=credit+age:signature=E7AHv9be7f4Tnc5d+C7CFQ is the permit.

In response to receiving the delegation request (including the permit) 52, the service provider verifies the user at step 54 of the algorithm 40. The step 54 implements steps 14 and 16 of the algorithm 10, and may also implement step 18 (if appropriate). Thus, in response to receiving the message 52, the service provider 6 extracts the user’s password based on the username, then verifies the permit signature using the key generated from the password.

As described above with reference to Figure 2, the service provider first extracts the ‘permit’ field in message 52 and finds the username (“jerry01”). The service provider then finds the key stored for the user and generates a signature using a pre-defined algorithm like HMAC-SHA-1 based on the permit string
"uid=jerry01:site=sp.com:attributelist=credit+age", excluding the signature part. The generated signature can then be BASE64 encoded and compared to the one in the request message (E7AHr9be7f4Tnc5d+C7CFQ==). The verification is considered passed if the two string equal to each other, otherwise it fails.

If the user is verified at step 54, then requested resources (such as the requested user data - credit and age in this example) are provided by the service provider 6 to the application 4 as message 56.

It should be noted that once the application 4 is in possession of a valid permit, the message steps 52 to 56 can be repeated to enable repeated access to data stored at the service provider 6, if the permit allows repeated access. As mentioned above, the permit may include an expiry time, so that repeated access may be allowed for a limited period of time.

The message sequence 60 shown in Figure 6 starts with the user 2 logging in to a profile page at the service provider 6, as indicated by the arrow 62. Once logged in, the user 2 defines user policies at the service provider 6, as indicated by the box 64, thereby implementing the step 32 described above. The policies can be used to define an attribute sharing policy, for example by enabling the user to select a particular IDM for delegation. Thus, the user 2 can delegate user attribute access to the application 4.

Next, at step 66, the user 2 indicates to the service provider 6 that delegation permission is to be given to the application 4, thereby implementing the step 34. In response to the message 66, the service provider 6 creates a permit at step 68, thereby implementing the step 36. The permit may take the form described above.
The permit is sent to the application 4 via the user 2 by redirection in a step 70 (although, as discussed above, the permit may be sent directly to the application 4 in some embodiments of the invention). Thus, at this stage, the user 2 has defined delegation permissions and those permissions have been implemented by providing the application 4 with the required permit.

As shown in Figure 6, the user 2 next optionally logs in at the application 4 as shown by the message 72. (The step 72 could be omitted, since the application 4 could request access without the user being logged in to the application.)

The application 4 wants to access user attributes or some other data concerning the user that is stored at the service provider 6 and sends a message 74, including the permit discussed above, to the service provider 6, thereby implementing the step 12. The request 74 and may take the same form as the request 52 described above.

In response to the request 74, the service provider 6 verifies the user at step 76 (implementing steps 14 and 16), which step may be implemented in the same manner as the step 54. Finally, the request resources or data are provided to the application 4 is step 78.

The present invention seeks to provide a lightweight, convenient, practical and efficient solution to the problem of sharing user data. Some important features of the invention are set out below. It should be noted that, unless otherwise stated, none of the following features should be considered to be essential to all embodiments of the invention, and the invention encompasses embodiments including any combination of the features discussed below.

The solution provided by the present invention does not significantly increase the processing burden on either the
service provider 6 or the application 4 due to the delegation operation.

Most service providers provide application programming interfaces (APIs) in the format a uniform resource locator (URL), through which the retrieval operation is done with a single HTTP request/response. Other messages exchanged between user 2, application 4 and service provider 6 are overhead and are kept to a minimum.

The application 4 and the service provider 6 are decoupled to a significant degree. Considering the large number of service providers on the Internet, any requirements for signing contracts or manual registration is unscalable and thus should generally be avoided.

End users have control over what information at the service provider 6 can be revealed to the application 4, before the application interacts with the service provider on behalf of the user.

By using a permit system, the service provider 6 is able to determine whether the end user or his delegator (the application 4) is making the request, which enables the service provider to provide appropriate service and save meaningful data in a log for later audit.

The solution is not dependent on the form of the terminal used (such as personal computer, mobile communication devices, set-top box etc.), provided that the terminal concerned includes a web browser.

The embodiments of the invention described above are illustrative rather than restrictive. It will be apparent to those skilled in the art that the above devices and methods may incorporate a number of modifications without departing from the general scope of the invention. It is intended to include all such modifications within the scope of the
invention insofar as they fall within the scope of the appended claims.
CLAIMS:

1. A method comprising:
   receiving a delegation authorisation request identifying
   a resource for which delegated authorisation is requested;
   determining whether or not the delegation authorisation
   request is authorised by a user; and
   creating a delegation permit, the permit defining
   information for which delegated authorisation is granted.

2. A method as claimed in claim 1, wherein the permit
   includes one or more of a username of the user at a service
   provider including said resource, a policy identifying one or
   more resources for which delegated authorisation is granted
   and a digital signature.

3. A method as claimed in claim 1 or claim 2, further
   comprising signing the delegation permit using a key based on
   a value specific to the user.

4. A method as claimed in any one of claims 1 to 3, wherein
   determining whether or not the delegation authorisation
   request is authorised by the user includes presenting a
   prompt to the user.

5. A method as claimed in any preceding claim, wherein the
   delegation authorisation request is received from an
   application requesting delegated authorisation.

6. A method as claimed in any one of claims 1 to 4, wherein
   the delegation authorisation request is received from the
   user.

7. A method as claimed in claim 6, wherein determining
   whether or not the delegation authorisation request is
   authorised by the user includes determining that the request
   is received from the user.
8. A method as claimed in any preceding claim, further comprising:
   receiving, at a service provider, a request for delegated authorisation to access a resource, wherein the request includes the said permit; and
   checking the details of the permit to determine whether delegated authorisation to access said resources has been granted.

9. A method comprising:
   receiving at a service provider, an access request from an application, wherein the request includes a permit;
   checking the details of the permit to determine whether the requested access is authorised; and
   providing the requested access in the event that access is deemed to be authorised.

10. A method as claimed in claim 9, wherein the permit includes one or more of a username of a user at the service provider, a policy identifying one or more resources for which delegated authorisation is granted and a digital signature.

11. A method as claimed in claim 9 or claim 10, wherein checking the details of the permit to determine whether the access request is authorised comprises:
   extracting an identity for the user from the permit; and
   using the user identity to determine a key used to sign the permit from the user in order to verify a permit signature.

12. An apparatus comprising:
   a first input for receiving a delegation authorisation request identifying a resource for which delegated authorisation is requested;
   a first processor for determining whether or not the delegation authorisation request is authorised by a user;
a second processor adapted to create a delegation permit, wherein the delegation permit includes one or more of a username of the user at a service provider including said resource, a policy identifying one or more resources for which delegated authorisation is granted and a digital signature; and

a first output for outputting the delegation permit.

13. An apparatus as claimed in claim 12, further comprising a second output for presenting a prompt to the user requesting an indication of whether the delegation authorisation request is authorised by the user and a second input for receiving a response to the prompt, wherein the first processor determines whether or not the delegation authorisation request is authorised by a user on the basis of said response to said prompt.

14. An apparatus as claimed in claim 12 or claim 13, further comprising:

a third input for receiving a request for access to one or more resources, wherein the request includes the said permit; and

a third processor for checking the details of the permit to determine whether delegated authorisation to access said resources has been granted.

15. An apparatus comprising:

a first input for receiving an access request from an application, wherein the request includes a permit;

a first processor adapted to check the details of the permit to determine whether access to the service is authorised; and

a second processor adapted to provide access to the service in the event that access is deemed to be authorised.

16. An apparatus as claimed in claim 15, wherein the first processor is adapted to:

extract an identity for the user from the permit; and
use the user identity to determine a key used to sign the permit from the user in order to verify a permit signature.

17. A delegation authorisation permit, the permit comprising:
   a username for a user at a service provider;
   a policy identifying resources stored at the service provider for which delegated authorisation is granted; and
   a digital signature,
   wherein the signature is calculated using a key generated from a value specific to the user.

18. A computer program product comprising:
   means for receiving a delegation authorisation request identifying a resource for which delegated authorisation is requested;
   means for determining whether or not the delegation authorisation request is authorised by a user; and
   means for creating a delegation permit, the permit defining information for which delegated authorisation is granted.

19. A computer program product comprising:
   means for receiving at a service provider, an access request from an application, wherein the request includes a permit and wherein the request requests access to a resource;
   means for checking details of the permit to determine whether access to the resource is authorised; and
   means for providing access to the resource in the event that access is deemed to be authorised.
Fig. 5
**INTERNATIONAL SEARCH REPORT**

**A. CLASSIFICATION OF SUBJECT MATTER**

INV. H04L29/06 H04L12/22

ADD.

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

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

H04L G06F

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 practical, search terms used)

EPO-Internal

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
</table>

- Special categories of cited documents:
  - "A" document defining the general state of the art which is not considered to be of particular relevance
  - "E" earlier document 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
- "A" document member of the same patent family

Date of the actual completion of the international search: 13 July 2010

Date of mailing of the international search report: 21/07/2010

Name and mailing address of the ISA/
European Patent Office, P. B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk
Tel: (+31-70) 340-9040. Fax: (+31-70) 340-3016

Authorized officer: Veen, Gerardus

Form PCT/ISA/210 (second sheet) (April 2005)
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>US 7185047 B1</td>
<td>27-02-2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2007261106 A1</td>
<td>08-11-2007</td>
</tr>
</tbody>
</table>

---

Form PCT/ISA/2bis (patent family annex) (April 2008)