Fig. 5

(54) Title: GBA AND IMS AUTHENTICATION PROCEDURES

(57) Abstract: A method of combining GBA and IMS authentication procedures is described. The invention enables a user to be authenticated in an IMS system and to provide user identities applicable to the IMS session to an application as part of a GBA authentication process, such that the application can communicate with the user using IMS/SIP protocols, with confidence that the said user identities really belong to the served user and can be used to contact them.
GBA and IMS Authentication Procedures

The present invention relates to the user identification, in particular to user identification and authentication in GBA and IMS systems.

Generic Bootstrapping Architecture (GBA) is a known solution for re-using telecommunication authentication, typically for end-user authentication in Internet applications. Figure 1 shows a system, indicated generally by the reference numeral 1, showing key elements of a GBA system.

The system 1 comprises a user device or user equipment (UE) 2, bootstrapping server function (BSF) 4, home subscriber server (HSS) 6 and a network application function (NAF) 8. The UE 2, BSF 4 and HSS 6 all form part of a home network and the NAF 8 is a part of a visited network.

In the use of the system 1, the user device 2 is seeking access to the network application function (NAF) 8. Before the NAF 8 can grant the requested access to the user device 2, the user must be authenticated. This is achieved using the known GBA bootstrapping procedure, which is briefly described below.

Figure 2 shows a message sequence, indicated generally by the reference numeral 10, showing, in general terms, an exemplary use of the GBA bootstrapping procedure to enable the user device 2 to gain access to a service provided by the NAF 8. As will be known to persons skilled in the art, the message sequence 10 is a simplification of the GBA bootstrapping procedure, as set out, for example, in the 3GPP specification TS33.220.
The message sequence 10 begins with the user device 2 sending an initial request 11 to the NAF 8. At this stage, the user device 2 is unknown to the NAF 8 and so the request is not accepted. Thus, an unauthorised response 12 is sent from the NAF 8 to the user device 2. The unauthorised response 12 indicates to the user device that GBA should be used to authenticate the user at the application 8.

In response to the unauthorised response 12, a further request 13 is sent from the user device 2 to the BSF 4. The request 13 includes the user's IMPI (IP Multimedia Private Identifier), extracted from the subscriber identification module (SIM) hosted by the smartcard (UICC) residing on user device 2. The BSF 4 sends a message 14 to the HSS 6 requesting an authentication vector and GBA user security settings (GUSS) corresponding to the IMPI provided by the user device 2. The HSS 6 returns the requested authentication vector and GUSS to the BSF 4 in a message 16.

The user is not, at this stage, authenticated and so the BSF 4 responds to the request 13 by sending an unauthorised message 18 to the user device 2. The unauthorised message 18 includes a random number (RAND) included in the authentication vector obtained from the HSS 6.

At step 20, the user device 2 uses data stored on its SIM to apply the AKA (Authentication and Key Agreement) algorithm on the random number RAND to generate a response (RES) and session keys CK and IK.

Once the response RES has been generated by the user device 2, the user device generates a third request 22 and sends that request to the BSF 4. The third request 22 includes derived response RES.
On receipt of the third request 22, the BSF 4 compares the response RES with an expected response XRES that was provided as part of the authentication vector included in the message 16. If the response RES matches the expected response XRES, then the user is authenticated. If the user is authenticated, then, at step 23 of the algorithm 10, the BSF 4 generates a bootstrapping transaction identifier (B-TID) and sends an OK message 24, including the identifier B-TID, to the user device.

At this stage in the algorithm 10, the user device 2 has been authenticated at the BSF 4 and both the user device 2 and the BSF 4 have all the information that they require in order to generate session keys for authentication purposes. The next step of the algorithm 10 is for the user device 2 to make use of the authentication.

At step 26 of the algorithm 10, the user device 2 requests access to an application provided by the NAF 8. The application request 26 includes the B-TID provided to the user device by the BSF 4.

In response to the application request 26, the NAF 8 sends a bootstrapping information request 28 to the BSF 4 in order to obtain the key material required to provide the user device with access to the NAF. The BSF responds to the request 28 by sending a bootstrapping information answer 30 to the NAF 8. The NAF 8 then grants the user device access to the requested application by sending an application response message 32 to the user device 2.

The authorisation message 24 may include an indication of a period of time over which the authorisation is valid. The
steps 26 to 32 can be repeated (with the same or other NAFs) for the period of validity of the authentication.

IP Multimedia Subsystem (IMS) is an architectural framework for delivering Internet Protocol (IP) multimedia services. IMS uses Session Initiation Protocol (SIP), which is a signalling protocol that is used for initialising and ending multimedia communication sessions such as voice and video calls over the Internet.

Figure 3 is a block diagram, indicated generally by the reference numeral 40, showing an exemplary IMS system. The IMS system 40 comprises a user device (UE) 42, a call session control function (CSCF) server 44, a home subscriber server (HSS) 46 and an application server (AS) 48.

The user device 42 and the user of that user device have a number of identities. These include the well known IMSI, TMSI, IMEI and MSISDN numbers. In addition, a user has an IP Multimedia Private Identity (IMPI) and one or more IP Multimedia Public Identities (IMPUs). The HSS 46 stores a variety of data regarding the user, including IMSI, MSISDN, IMPI and IMPU data.

In an exemplary use of the system 40, the user device 42 seeks access to the application server 48. The CSCF 44 and HSS 46 are used to authenticate the user.

As will be understood by the skilled person, the arrangement shown in Figure 3 is a simplification of a real IMS system. For example, the CSCF module 44 comprises a proxy-CSCF module (which is generally the first point of contact for the user device 42 with the IMS system), a serving-CSCF and an interrogating-CSCF.
Figure 4 is a flow chart, indicated generally by the reference numeral 50, showing, in very broad terms, an exemplary use of the system 40. The flow chart starts at step 52, where the IMS device 42 is registered with the IMS system. The step 52 is achieved using a SIP REGISTER message. Once registered, the user device 42 can initiate a session at the application server 48 at step 54. An important part of the IMS registration is that a subset of the user's IMPUs is registered as part of the IMS session; this means that from then on, the user can be reached via any of the registered IMPUs.

Thus, GBA and IMS both offer user authentication procedures. Furthermore, GBA is based on the same authentication mechanism (AKA) that IMS is. However, the two "worlds" otherwise remain separated. For example, when a GBA-enabled web site (NAF) authenticates a user, it has no further knowledge regarding the user's identifiers used in the IMS. In many cases, this is desirable; there are other cases, however, when it might be desirable for user's IMS identifiers (IMPUs) to be communicated to the GBA-enabled web site (as discussed further below).

The present invention seeks to address at least some of the problems outlined above.

The present invention provides a service provider, wherein the service provider includes a resource and wherein the service provider requires a user requesting access to the resource to be authenticated using generic bootstrapping architecture, the service provider comprising: a first input for receiving a request to access the resource from a first user device, wherein the first user device is referred to the resource by a second service provider (typically in response to
a request from a second user device); and means for obtaining one or more registered IP multimedia public identities (IMPU(s)) for the first user device as registered for an IMS communication session between the first user device and a second user device (such that, for example, the service provider is (later) able to communicate with the first user device using the IMS protocol).

The present invention also provides a method comprising: receiving a request at a first service provider for access to a resource provided by the first service provider, wherein the request is received from a first user device and wherein the first user device is referred to the resource by a second service provider (typically in response to a request from a second user device); and authenticating the first user device at the first service provider in accordance with generic bootstrapping architecture protocol, wherein authenticating the first user device includes obtaining one or more registered IP multimedia public identities (IMPU(s)) for the first user device as registered for an IMS communication session between the first user device and a second user device (such that, for example, the first service provider is (later) able to communicate with the first user device using the IMS protocol).

Accordingly, the present invention provides a service provider and a message that combine GBA and IMS authentication procedures. The invention enables a user to be authenticated in an IMS system and to provide user identities applicable to the IMS session to an application as part of a GBA authentication process, such that the application can communicate with the user using IMS/SIP protocols, with confidence that the said user identities really belong to the served user and can be used to contact them.
In some forms of the invention, the one or more registered IP multimedia public identities are received at the service provider in a request received at said first input from the first user device requesting access to the resource. The service provider may obtain a list of all IP multimedia public identities for said first user device (both registered and unregistered) and compare the IP multimedia identities included in the list with the one or more registered IP multimedia identities included in the request received from the first user device. Such an arrangement seeks to prevent a service provider from being tricked into contacting another user in an unsolicited way. In one form of the invention, the one or more registered IP multimedia public identities for the first user device are stored at the first user device during an IMS registration process.

In some forms of the invention, the one or more registered IP multimedia public identities are obtained by the service provider. For example, the service provider may be adapted to contact a home subscriber service for the first user device in order to obtain said one or more registered IP multimedia public identities. Alternatively, the service provider may be adapted to contact a bootstrapping server function for the first user device in order to obtain said one or more registered IP multimedia public identities. In yet a further alternative arrangement, a home subscriber server for the first user device may be arranged to generate (typically dynamically) a list of registered IP multimedia public identities (for example from a template or by some other means) for the first user device during the GBA bootstrapping procedure, and the service provider may be adapted to obtain said one or more IP multimedia public identities obtained by requesting said list.
In many forms of the invention, the second user device instructs the first user device to contact the service provider by means of an SIP REFER message.

The present invention further provides a user device comprising: a first input for receiving a referral message (typically an SIP REFER message) instructing the user device to access a resource at a first service provider, wherein the first service provider is a network application function that requires the first user device to be authenticated using a Generic Bootstrapping Architecture protocol; and a first output for sending a request for access to the resource at the first service provider, wherein the user device is adapted to provide (to the first service provider) one or more registered IP multimedia public identities (IMPU(s)) for the first user device as registered for an IMS communication session between the first user device and a second user device, such that the first service provider is (later) able to communicate with the user device using the IMS protocol. The user device may include the one or more registered IP multimedia public identities in the request for access to the resource at the first service provider.

The present invention also provides a method comprising: receiving a referral (typically an HTTP referral) from a second service provider at a first user device, wherein the referral refers the first user device to resource at a first service provider, wherein the second service provider requires the first user device to be authenticated using GBA authentication; the first user device requesting access to the resource at the second service provider; and providing (to the first service provider) one or more IP multimedia public identities (IMPU(s)) for the first user device as registered for an IMS
communications session between the first user device and a second user device to the second service provider as part of a GBA authentication procedure at the first service provider.

The present invention also provides a system comprising a first user device, a first service provider and a second service provider, wherein: the first user device is adapted to communicate with a second user device via the second service provider, wherein the second service provider is an IMS application server; the second service provider is adapted to instruct the first user device to contact the first service provider, wherein the first service provider is a network application function that requires the first user device to be authenticated using Generic Bootstrapping Architecture; and the first service provider receives one or more IP multimedia public identities for the first user device as registered for the IMS communication between the first and second user devices.

The present invention yet further provides a method comprising referring a first user device that is active in an IMS session with a second service provider to a resource, wherein the resource is at a first service provider, wherein the first service provider is a network application function that requires the first user device to be authenticated using Generic Bootstrapping Architecture in order to access the resource and the second service provider is an IMS application server, the method further comprising: issuing or receiving a referral from the second service provider to the first user device instructing the first user device to access the said resource at the first service provider; conducting GBA authentication at the first service provider; and obtaining one or more registered IMPUs for the first user device as part of
the GBA authentication procedure conducted at the first service provider.

The present invention also provides computer program comprising: code (or some other means) for receiving a request at a first service provider for access to a resource provided by the first service provider, wherein the request is received from a first user device and wherein the first user device is referred to the resource by a second service provider (typically in response to a request from a second user device); and code (or some other means) for authenticating the first user device at the first service provider in accordance with generic bootstrapping architecture protocol, wherein authenticating the first user device includes obtaining one or more registered IP multimedia public identities (IMPU(s)) for the first user device as registered for an IMS communication session between the first user device and a second user. 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.

The present invention also provides a computer program comprising: code (or some other means) for receiving a referral (typically an HTTP referral) from a second service provider at a first user device, wherein the referral refers the first user device to resource at a first service provider, wherein the second service provider requires the first user device to be authenticated using GBA authentication; code (or some other means) to enable the first user device to request access to the resource at the second service provider; and code (or some other means) for providing one or more IP multimedia public identities (IMPU(s)) for the first user device as registered for an IMS communications session between the first user device and a second user device to the second service.
provider as part of a GBA authentication procedure at the first service provider. 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.

The present invention yet further provides a computer program comprising code (or some other means) for referring a first user device that is active in an IMS session with a second service provider to a resource, wherein the resource is at a first service provider, wherein the first service provider is a network application function that requires the first user device to be authenticated using Generic Bootstrapping Architecture in order to access the resource and the second service provider is an IMS application server; code (or some other means) for issuing or receiving a referral from the second service provider to the first user device instructing the first user device to access the said resource at the first service provider; conducting GBA authentication at the first service provider; and code (or some other means) for obtaining one or more registered IMPUs for the first user device as part of the GBA authentication procedure conducted at the first service provider. 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 numbered schematic drawings.

Figure 1 is a simplified block diagram of a known GBA system;
Figure 2 is a message sequence showing an exemplary use of the GBA system of Figure 1;
Figure 3 is a simplified block diagram of a known IMS system;
Figure 4 is a flow chart showing an exemplary use of the IMS system of Figure 3;
Figure 5 is a block diagram showing an exemplary scenario in which the present invention may be used; and
Figures 6 to 18 are flow charts showing algorithms in accordance with various aspects of the present invention.

Figure 5 is a block diagram, indicated generally by the reference numeral 60, showing a scenario in which the present invention may be used. The block diagram 60 shows a user device for a first user (Alice) 62, a user device for a second user (Bob) 64, an application server (AS) 65 and a service provider (SP) 66. The application server 65 is in two-way communication with both the user device 62 and the user device 64. Similarly, the service provider 66 is also in two-way communication with both the user device 62 and the user device 64.

In one form of the invention, the application server 65 provides a video conference system that enables and controls communications between Alice and Bob using IMS and the service provider 66 provides a photo sharing service that makes use of GBA for authentication. Of course, the video conferencing and photo sharing service are two of many applications that might be provided by the application server 65 and the service provider 66.

Consider the following exemplary scenario. Alice uses the IMS-enabled user device 62 to initiate a conversation (over SIP) with Bob, using the application server 65, where Bob is
using the IMS-enabled device 64. Accordingly, Alice and Bob are authenticated in accordance with the IMS protocol. As part of the conversation, Alice wants to show Bob a photo which is available at the service provider 66. As noted above, the service provider 66 makes use of GBA to perform user authentication such that both Alice and Bob are authenticated at the service provider 66 using GBA.

Figure 6 is a flow chart showing, in broad terms, an algorithm, indicated generally by the reference numeral 70, showing an exemplary use of the system 60 to implement the scenario described above. The algorithm 70 starts at step 72, with an IMS initiation procedure, in which SIP communications are set up under the control of the application server 65 between Alice (the user device 62) and Bob (the user device 64). The step 72 uses conventional IMS registration steps that are essentially the same as the steps 52 and 54 described above with reference to Figure 4.

Next, the algorithm 70 moves to step 74, where Alice wants to inform Bob of a photo that he should look at. This step is achieved using a conventional SIP referral that is sent from the application server 65 to the user device 64 for Bob, inviting him to access the photo store at the application 66. The step 74 may be implemented with the following SIP message:

REFER sip:user.name@op.com SIP/2.0

...  

Refer-To: http://sp.com/abc
Content-Length: 0

Where the photograph that Bob is being informed about can be found at the URL http://sp.com/abc.
The algorithm 70 then moves to step 76, where Bob seeks to access the photograph and where a conventional GBA authorisation step is carried out (for example, as described above with reference to Figure 2), such that Bob is provided with access to the photo store at the application 66. (Of course, Alice may also have been authenticated at the service provider 66; this step is not shown in the algorithm 70.)

Once Bob has been authenticated at the service provider 66 using the GBA authentication procedure, Bob can view the photo referred to him by Alice.

Assume now that the service provider 66 wants to initiate an alternative multimedia session with Bob (using IMS). For example, the service provider 66 might want to send an instant message or a short message to Bob to indicate that a new photograph has been uploaded to a particular folder on the photo application. Bob has already been authenticated by IMS (in order to communicate via the application server 65 with Alice) so an IMS communication is technically possible; but the relevant identification information (i.e. one of Bob's IMPUs) has not been communicated to the service provider 66.

In order to be able to initiate the IMS-based communication towards Bob, the photo sharing service (the service provider 66) needs to obtain Bob's public IMS identifiers (registered IMPU(s)). As described further below, the present invention enables the service provider 66 to obtain Bob's IMS identifiers as part of the GBA authentication procedure.

Figure 7 is a flow chart of an algorithm, indicated generally by the reference numeral 80, enabling the service provider to receive the relevant IMPUs for the user, as registered as
part of the IMS registration process, thereby enabling the
service provider 66 to initiate communication with Bob using
the IMS protocol.

5 The algorithm 80 starts at step 82, with an IMS registration.
The step 82 is a conventional IMS registration step, involv-
ing a SIP REGISTER message and is identical to the step 72
referred to above.

10 Next, the algorithm 80 moves to step 84, where an HTTP refer-
ral is sent from the application server 65 to the user of the
user device 64 (Bob), referring Bob to a resource at a ser-
vice provider (such as the photo store at the service pro-
vider 66). The service provider requires a user to be au-
thenticated using GBA before access can be granted and Bob is
not (at this stage) authenticated.

The step 84 differs from the step 74 discussed above in that
the REFER message includes details of the IMPUs for the user
Bob. The REFER step 84 may be as follows:

REFER sip:user.name@op.com SIP/2.0

... Refer-To: http://sp.com/abc?impul=sip :user.name@op.com

Content-Length: 0

Where the URL "http://sp.com/abc?impul=sip :user.name@op.com"
includes details of one or more IMPUs for Bob (one in this
case). Accordingly, the IMPU(s) for Bob are embedded in the
REFER message.

The algorithm 80 then moves to step 86, where Bob seeks to
access the photograph and where a GBA authorisation step is
carried out.
Figures 8 and 9 are flow charts (indicated generally by the reference numerals 90 and 100 respectively) that together show an algorithm showing an exemplary implementation of the GBA authentication step 86 described above. The algorithm shown by the flow charts 90 and 100 is similar to the algorithm 10 described above with reference to Figure 2, but differs in a number of important respects.

The flow chart 90 begins with the user device 64 (the user device for Bob) sending a first request 91 to the service provider 66 for access to the service referred to in the REFER message 84. At this stage, the user device 64 is unknown to the service provider and so the request is not accepted. Thus, an unauthorised response is sent from the service provider to the user device 64 (see message 12 of Figure 2). The unauthorised response indicates to the user device that GBA authentication should be used to authenticate the user at the service provider.

The first request 91 is issued by an HTTP client and includes details of the IMPUs for the user device 64 included in the HTTP referral 84. The first request 91 may take the form of an HTTP GET command as follows:

```
HTTP GET /abc?impul=sip :user .name@op .com&
    impu2=sip :+358504836551@op.com; user=phone
```

Thus, the exemplary HTTP GET command above includes two registered IMPUs for the user device 64 (labeled "impul" and "impu2" above). This, of course, assumes that the REFER message 84 included those two IMPUs (and not a single IMPU as set out in the example above).
In response to the unauthorised response, a request 92 is sent from the user device 64 to the BSF associated with the user device 64. The request 92 includes the user's IMPI (IP Multimedia Private Identifier), extracted from the subscriber identification module (SIM) hosted by the smartcard (UICC) residing on user device 64. The BSF sends a message to the HSS requesting an authentication vector and GBA user security settings (GUSS) corresponding to the IMPI provided by the user device 64 (see message 14 of Figure 2). The HSS returns the requested authentication vector and GUSS to the BSF in a message 93.

The GUSS returned by the HSS to the BSF is extended (when compared with the GUSS included in the message 16) to include IMPUs for the user device. An exemplary form for the GUSS might be as follows:

```
<guss id="35850 0004836551@ims .mnc050 .mcc358 .3gppnetwork .org">
  <bsfInfo>. ..</bsfInfo>
  <ussList>
    <uss id="1" type="0" nafGroup=" ...">
      <uids>
        <uid>donald_duck</uid>
        <uid>sip :user .name @op .com</uid>
        <uid>sip:+358504836551@op .com; user=phone</uid>
      </uids>
      <flags></flags>
    </uss>
  </ussList>
...</guss>
```

Thus, the GUSS also includes both the registered and unregistered IMPUs for the user device 64. The above shown GUSS con-
tains more than one <uid> elements. The first one of them, as usual in GBA, contains the NAF-specific user identifier i.e. the identifier by which the NAF (identified by the nafGroup attribute) "knows" the user. The further <uid> elements contain IMPUs. The most straightforward use of this list is where the BSF sends all of the user's IMPUs (both registered and unregistered) to the NAF when requested to do so (cf. arrows 28 and 30 in Figure 2), but more sophisticated cases are possible (as discussed further below).

The user is not, at this stage, authenticated and so the BSF responds to the request 92 by sending an unauthorised message (similar to the message 18 of Figure 2) to the user device 64. The unauthorised message includes a random number (RAND) included in the authentication vector obtained from the HSS.

At step 94, the user device 64 uses data stored on its SIM to apply the AKA algorithm on the random number RAND to generate a response (RES) and session keys CK and IK.

Once the response RES has been generated by the user device 64, the user device generates a request and sends that request to the BSF (message 96). The request 96 includes derived response RES.

On receipt of the request 96, the BSF compares the response RES with an expected response XRES that was provided as part of the authentication vector included in the message 93. If the response RES matches the expected response XRES, then the user is authenticated (step 97). If the user is authenticated, then the BSF generates a bootstrapping transaction identifier (B-TID) and sends an OK message, including the identifier B-TID, to the user device.
At this stage in the algorithm, the user device 64 has been authenticated at the BSF and both the user device 64 and the BSF have all the information that they require in order to generate session keys for authentication purposes. The next step of the algorithm (after the step 97 referred to above) is for the user device 64 to make use of the authentication. This part of the algorithm is shown in the flow chart 100.

The flow chart 100 starts at step 102, where the user device 64 once again requests access to the service provider 66. The application request 102 includes the B-TID provided to the user device by the BSF.

In response to the application request 102, the service provider 66 sends a bootstrapping information request to the BSF in order to obtain the key material required to provide the user device with access to the service provider (see message 28 of Figure 2). The BSF responds by sending a bootstrapping information answer to the service provider (see message 30 of Figure 2). The service provider then grants the user device access to the requested application by sending an application response message 104 to the user device 64.

With the user device 64 authorised using the GBA procedure at the service provider 66, the algorithm 100 continues by comparing (at step 106) the IMPUs included in the original request 91 sent from the user device 64 to the service provider 66 with the IMPUs included in the GUSS sent from the HSS to the BSF at step 93. The GUSS includes a list of all potential IMPUs for the user device. The IMPUs included in the message 91 should therefore appear in the GUSS. This check counters the threat that a malicious user device could otherwise include any IMPU (e.g. another user's IMPU) into the original request 91, thereby preventing a service provider
from being tricked into contacting another user in an unsolicited way.

At step 108, the user device is granted access to the service provider 66. At this stage, the service provider is also aware of the IMPUs for the user device and is therefore able to communicate with the user device 64 via IMS. Accordingly (in the exemplary use of the invention described above), the service provider 66 can send a message to the user device informing the user (Bob) that a new photo has been uploaded to a particular folder, as described above.

Figure 10 is a block diagram showing an algorithm, indicated generally by the reference numeral 110, that is a variant of the algorithm 80 described above with reference to Figure 7.

The algorithm 110 starts at step 112, with an IMS registration. The step 112 is a conventional IMS registration step, involving a SIP REGISTER message and is identical to the steps 72 and 82 referred to above.

Next, the algorithm 110 moves to step 114, where the user device 64 notes the IMPUs that have been registered as part of the IMS initiation process.

With the IMPUs noted at the user device 64, the algorithm 110 moves to step 116, where an HTTP referral is sent (for example by the application server 65 to the user device 64) to refer a user (such as Bob) to a resource at a service provider (such as the photo store at the service provider 66). The service provider requires a user to be authenticated using GBA before access can be granted and Bob is not (at this stage) authenticated.
In the algorithm 110, the referral step 116 is identical to the step 74 described above with reference to Figure 6 and therefore differs from the step 84 in that the IMPUs for the user (Bob) are not included in the REFER message.

The algorithm 110 then moves to step 118, where Bob seeks to access the photograph and where a GBA authorisation step is carried out. The GBA authentication step 118 is implemented using the algorithm 90 described above.

As described above, the first request 91 of the algorithm 90 used to implement the step 118 may take the form of an HTTP GET command as follows:

```
HTTP GET /abc?impul=sip :user .name@op .com&
  impu2=sip :+358504836551@op.com; user=phone
```

The difference between the algorithms 80 and 110 is the source of the registered IMPUs that are provided in the request 91. In the algorithm 80, the registered IMPUs are provided in the REFER message 84: in the algorithm 110, the registered IMPUs are stored at the user device during (or after) the IMS registration procedure (in step 114), and are added to the HTTP request 91 by the user device 64.

Figure 11 is a block diagram showing an algorithm, indicated generally by the reference numeral 120, that is a further variant of the algorithms 80 and 110 described above.

The algorithm 120 starts at step 122, with an IMS registration. The step 122 is a conventional IMS registration step, involving a SIP REGISTER message and is identical to the steps 72, 82 and 112 referred to above.
Next, the algorithm 120 moves to step 124, where an HTTP referral is sent to refer a user (such as Bob) to a resource at a service provider (such as the photo store at the service provider 66). The service provider requires a user to be authenticated using GBA before access can be granted and Bob is not (at this stage) authenticated.

In the algorithm 120, the REFER step 124 is identical to the REFER steps 74 and 116 described above and therefore differs from the REFER step 84 in that the IMPUs for the user (Bob) are not included in the REFER message. The algorithm 120 also differs from the algorithm 110 in that the registered IMPU(s) for the user (Bob) are not noted/stored at the user device 64 (i.e. step 114 of the algorithm 110 is omitted).

The algorithm 120 then moves to step 126, where Bob seeks to access the photograph and where a GBA authorisation step is carried out. However, unlike in the arrangements described above, the initial GBA authentication request does not include details of the registered IMPUs for the user (Bob).

Finally, the algorithm 120 ends at step 128, where the service provider 66 obtains the IMPUs from the HSS.

The steps 126 and 128 of the algorithm 120 can be implemented in similar manner to the steps 86 and 118 of the algorithms 80 and 110 as described above with reference to Figures 7 and 10. The steps 126 and 128 are described further below with reference to Figures 12 and 13.

Figures 12 and 13 are flow charts (indicated generally by the reference numerals 90' and 130 respectively) that together show an algorithm demonstrating an exemplary implementation.
of the steps 126 and 128 of the algorithm 120 described above.

The algorithm 90' is similar to the algorithm 90 described above and starts with the user device 64 (the user device for Bob) sending a first request 91' to the service provider 66 for access to the service referred to in the REFER message 124. At this stage, the user device 64 is unknown to the service provider and so the request is not accepted. Unlike the request 91 described above, the request 91' does not include details of the IMPUs for the user device 64.

In response to the unauthorised response, a request 92' is sent from the user device 64 to the BSF associated with the user device 64. The request 92' includes the user's IMPI (IP Multimedia Private Identifier), extracted from the subscriber identification module (SIM) hosted by the smartcard (UICC) residing on user device 64. The BSF sends a message to the HSS requesting an authentication vector and GBA user security settings (GUSS) corresponding to the IMPI provided by the user device 64 (see message 14 of Figure 2). The HSS returns the requested authentication vector and GUSS to the BSF in a message 93'.

The GUSS returned by the HSS to the BSF is extended (when compared with the GUSS included in the message 16) to include all possible IMPUs for the user device. An exemplary form for the GUSS might be similar to that described above with reference to Figure 8.

The user is not, at this stage, authenticated and so the BSF responds to the request 92' by sending an unauthorised message, including a random number (RAND) included in the authentication vector obtained from the HSS. In response, the
user device 64 uses data stored on its SIM to apply the AKA algorithm on the random number RAND to generate a response (RES) and session keys CK and IK (step 94').

5 Once the response RES has been generated by the user device 64, the user device generates a request and sends that request to the BSF (message 96'). The request 96' includes derived response RES.

10 On receipt of the request 96', the BSF compares the response RES with an expected response XRES that was provided as part of the authentication vector included in the message 93'. If the response RES matches the expected response XRES, then the user is authenticated (step 97'). If the user is authenticated, then the BSF generates a bootstrapping transaction identifier (B-TID) and sends an OK message, including the identifier B-TID, to the user device.

At this stage, the user device 64 has been authenticated at the BSF. The next step of the algorithm is for the user device 64 to make use of the authentication. This part of the algorithm is shown in the flow chart 130 (which is similar to the flow chart 100 described above).

20 The flow chart 130 starts at step 132, where the user device 64 once again requests access to the service provider 66. The application request 132 includes the B-TID provided to the user device by the BSF.

25 In response to the application request 132, the service provider 66 sends a bootstrapping information request to the BSF in order to obtain the key material required to provide the user device with access to the service provider (see message 28 of Figure 2). The BSF responds by sending a bootstrapping
information answer to the service provider (see message 30 of Figure 2). The service provider then grants the user device access to the requested application by sending an application response message 134 to the user device 64.

At this stage, the service provider 66 still does not know the registered IMPUs for the user device 64. Accordingly, at step 136, the service provider asks the HSS to provide the registered IMPUs. The algorithm 130 then compares the IMPUs obtained from the HSS in step 136 with the IMPUs included in the GUSS sent to from the HSS to the BSF at step 93' described above. The GUSS includes a list of all potential IMPUs for the user device. The service provider keeps only those IMPUs from the GUSS that are also obtained from the HSS (the rest of the IMPUs are "inactive" at the moment).

At step 138, the user device is granted access to the service provider 66. At this stage, the service provider is also aware of the registered IMPUs for the user device (obtained in step 136) and is therefore able to communicate with the user device 64 via IMS. Accordingly, the service provider 66 can send a message to the user device informing the user (Bob) that a new photo has been uploaded to a particular folder, as described above.

Figure 14 is a block diagram showing an algorithm, indicated generally by the reference numeral 140, that is a further variant of the algorithms 80, 110 and 120 described above.

The algorithm 140 starts at step 142, with an IMS registration. The step 142 is a conventional IMS registration step, involving a SIP REGISTER message and is identical to the steps 72, 82, 112 and 122 referred to above.
Next, the algorithm 140 moves to step 144, where an HTTP referral is sent to refer a user (such as Bob) to a resource at a service provider (such as the photo store at the service provider 66). The service provider requires a user to be authenticated using GBA before access can be granted and Bob is not (at this stage) authenticated.

In the algorithm 140, the step 144 is identical to the steps 74, 116 and 124 described above and therefore differs from the step 84 in that the IMPUs for the user (Bob) are not included in the REFER message. The algorithm 140 also differs from the algorithm 110 in that the registered IMPU(s) for the user (Bob) are not noted/stored at the user device 64 (i.e. step 114 of the algorithm 110 is omitted).

The algorithm 140 then moves to step 146, where Bob seeks to access the photograph and where a GBA authorisation step is carried out. As with the request 126 described above, the initial GBA authentication request does not include details of the registered IMPUs for the user (Bob).

Finally, the algorithm 140 ends at step 148, where the BSF obtains the IMPUs from the HSS. Accordingly, the algorithm 140 differs from the algorithm 120 in that the BSF obtains the registered IMPUs, rather than the service provider requesting that information from the HSS.

Figure 15 is a flow chart, indicated generally by the reference numeral 150, showing a variant of the flow chart 130 suitable for use with the algorithm 140 described above. As with the flow chart 130, the flow chart 150 begins at the end of the flow chart 90’ described above, where the GBA bootstrapping steps have been implemented, but the registered IM-
PUUs for the user device are unknown to the service provider 66.

The flow chart 150 starts at step 152, where the user device 64 once again requests access to the service provider 66. The application request 152 includes the B-TID provided to the user device by the BSF.

In response to the application request 152, the service provider 66 sends a bootstrapping information request to the BSF in order to obtain the key material required to provide the user device with access to the service provider (see message 28 of Figure 2). The BSF responds by sending a bootstrapping information answer to the service provider (see message 30 of Figure 2). The service provider then grants the user device access to the requested application by sending an application response message 154 to the user device 64.

At this stage, the service provider 66 still does not know the registered IMPUs for the user device 64. The service provider requests that the BSF obtains the registered IMPUs and the BSF obtains the requested IMPUs from the HSS at step 156. The algorithm 150 therefore differs from the algorithm 130 in that the registered IMPUs for the user device are requested from the HSS by the BSF rather than by the service provider as in the algorithm 130. The algorithm 150 then compares the IMPUs obtained from the HSS in step 158 with the IMPUs included in the GUSS sent to from the HSS to the BSF at step 93' described above. The GUSS includes a list of all potential IMPUs for the user device, whereas the IMPUs obtained from the HSS in step 93' are those that are currently registered at the IMS. The intersection of the two lists is computed in step 158, as a result of which those and only
those IMPUs that appear in both lists are transmitted to the NAF (in step 30 as shown in Figure 2).

At step 159, the user device is granted access to the service provider 66. At this stage, the service provider is also aware of the IMPUs for the user device and is therefore able to communicate with the user device 64 via IMS. Accordingly, the service provider 66 can send a message to the user device informing the user (Bob) that a new photo has been uploaded to a particular folder, as described above.

Figure 16 is a block diagram showing an algorithm, indicated generally by the reference numeral 160, that is a further variant of the algorithms 80, 110, 120 and 140 described above.

The algorithm 160 starts at step 162, with an IMS registration. The step 162 is a conventional IMS registration step, involving a SIP REGISTER message and is identical to the steps 72, 82, 112, 122 and 142 referred to above.

Next, the algorithm 160 moves to step 164, where an HTTP referral is sent to refer a user (such as Bob) to a resource at a service provider (such as the photo store at the service provider 66). The service provider requires a user to be authenticated using GBA before access can be granted and Bob is not (at this stage) authenticated.

In the algorithm 160, the referral step 164 is identical to the steps 74, 116, 124 and 144 described above and therefore differs from the step 84 in that the IMPUs for the user (Bob) are not included in the REFER message. The algorithm 160 also differs from the algorithm 110 in that the registered
The IMPU(s) for the user (Bob) are not noted/stored at the user device 64 (i.e. step 114 of the algorithm 110 is omitted).

The algorithm 160 then moves to step 166, where Bob seeks to access the photograph and where a GBA authorisation step is carried out. As with the requests 126 and 146 described above, the initial GBA authentication request does not include details of the registered IMPUs for the user (Bob).

Finally, the algorithm 160 ends at step 168, where the HSS obtains the IMPUs, as described further below.

Figures 17 and 18 are flow charts (indicated generally by the reference numerals 170 and 180 respectively) that together show an algorithm demonstrating an exemplary implementation of the steps 166 and 168 of the algorithm 160 described above.

The algorithm 170 is similar to the algorithm 90 and 90' described above and starts with the user device 64 (the user device for Bob) sending a first request 171 to the service provider 66 for access to the service referred to in the REFER message 124. At this stage, the user device 64 is unknown to the service provider and so the request is not accepted. Unlike the request 91 described above, the request 171 does not include details of the IMPUs for the user device 64.

In response to the unauthorised response, a request 172 is sent from the user device 64 to the BSF associated with the user device 64. The request 172 includes the user's IMPI (IP Multimedia Private Identifier), extracted from the subscriber identification module (SIM) hosted by the smartcard (UICC) residing on user device 64. The BSF sends a message to the
HSS requesting an authentication vector and GBA user security settings (GUSS) corresponding to the IMPI provided by the user device 64 (see message 14 of Figure 2). The HSS returns the requested authentication vector and GUSS to the BSF in a message 173.

The response 173 differs from similar responses described above in that the GUSS generated and included in the response is a dynamic GUSS generated from a template.

The GUSS template, when being stored in the HSS, may take the following form:

```
<guss id="35850 0004836551@ims.mnc050.mcc358.3gppnetwork.org">
  <bsfInfo>...</bsfInfo>
  <ussList>
    <uss id="l" type="0" nafGroup=" ...">
      <uids>
        <uid>donald_duck</uid>
      </uids>
    </uss>
  </ussList>
</guss>
```

The template contains a "dynamic" part (enclosed by the "<?" and "?>" tags in the above example). This "dynamic" part is
evaluated by the HSS at the time the GUSS is being retrieved by the BSF (step 173). As a result, the GUSS returned to the BSF only contains the currently registered IMPUs of the user.

The user is not, at this stage, authenticated and so the BSF responds to the request 172 by sending an unauthorised message, including a random number (RAND) included in the authentication vector obtained from the HSS. In response, the user device 64 uses data stored on its SIM to apply the AKA algorithm on the random number RAND to generate a response (RES) and session keys CK and IK (step 174).

Once the response RES has been generated by the user device 64, the user device generates a request and sends that request to the BSF (message 176). The request 176 includes derived response RES.

On receipt of the request 176, the BSF compares the response RES with an expected response XRES that was provided as part of the authentication vector included in the message 173. If the response RES matches the expected response XRES, then the user is authenticated (step 177). If the user is authenticated, then the BSF generates a bootstrapping transaction identifier (B-TID) and sends an OK message, including the identifier B-TID, to the user device.

At this stage, the user device 64 has been authenticated at the BSF. The next step of the algorithm is for the user device 64 to make use of the authentication. This part of the algorithm is shown in the flow chart 180 (which is similar to the flow charts 100 and 130 described above).

The flow chart 180 starts at step 182, where the user device 64 once again requests access to the service provider 66.
The application request 182 includes the B-TID provided to the user device by the BSF.

In response to the application request 182, the service provider 66 sends a bootstrapping information request to the BSF in order to obtain the key material required to provide the user device with access to the service provider (see message 28 of Figure 2). The BSF responds by sending a bootstrapping information answer to the service provider (see message 30 of Figure 2). The service provider then grants the user device access to the requested application by sending an application response message 184 to the user device 64.

At this stage, the service provider 66 still does not know the registered IMPUs for the user device 64. Accordingly, at step 186, the service provider obtains the registered IMPUs, as included in the GUSS generated in step 173.

Finally, at step 188, the user device is granted access to the service provider 66. At this stage, the service provider is also aware of the registered IMPUs for the user device and is therefore able to communicate with the user device 64 via IMS. Accordingly, the service provider 66 can send a message to the user device informing the user (Bob) that a new photo has been uploaded to a particular folder, as described above.

The present invention seeks to enable an IMS-user device to provide IMS identifiers (IMPUs) to a service provider as part of a GBA authentication process at that service provider. Five different implementations of the invention are described above. The relative merits of the five implementations are discussed below.
In the first implementation (see Figures 7 to 9), registered IMPUs are sent to a service provider/NAF by the IMS-enabled user device as part of an HTTP request. The registered IMPUs are retrieved and sent to the user device by the IMS application service. The SP/NAF verifies IMPUs against a list of valid IMPUs received from the BSF (defined in HSS/GUSS). The first implementation requires the user device to be able to interact with both the SIP client and the HTTP client (browser) in order to pass IMPUs. The application server of the IMS system is required to retrieve the IMPUs and to pass the IMPUs to the user device.

The first implementation has the advantage that no modifications are needed to the network core infrastructure and is ideal for cases where HTTP requests are initiated by the application server. Disadvantages include the requirement to modify the user device (across many different device types) and the requirement for each application server to be able to send IMPUs.

In the second implementation (see Figure 10), registered IMPUs are sent to a service provider/NAF by the IMS-enabled user device as part of an HTTP request (as in the first implementation). The registered IMPU(s) are maintained by the user device (rather than being sent to the user device by the IMS application service). The service provider/NAF verifies IMPUs against a list of valid IMPUs received from the BSF (defined in HSS/GUSS). The second implementation requires the user device to be able to interact with both the SIP client and the HTTP client (browser) in order to pass IMPUs.

The second implementation has the advantage that no modifications are needed to the network core infrastructure. Disad-
vantages include the requirement to modify the user device (across many different device types).

In the third implementation (see Figures 11 to 13), a list of all possible IMPUs (both registered and un-registered) is provided in the GUSS. That list is filtered by the server provider/NAF. The service provider contacts the HSS in order to do this. The third implementation requires the service provider/NAF to be adapted to so it can filter the received IMPUs.

The third implementation has the advantages that no modifications are needed to the network core infrastructure and, unlike the first and second implementations, no modifications are required to the user devices. Disadvantages include the requirement that IMPU retrieval from the HSS must be provided in each service provider and in that the load on the HSS is increased. Also, the HSS may not allow the requested data to be provided to the service provider/NAF.

In the fourth implementation (see Figures 14 and 15), a list of all possible IMPUs (both registered and un-registered) is provided in the GUSS. That list is filtered by the BSF. The BSF (rather than the service provider/NAF as in the third implementation) contacts the HSS in order to do this. The fourth implementation requires the BSF to be adapted to so it can filter the received IMPUs.

The fourth implementation has the advantage that only the BSF requires any modification over existing elements. Disadvantages include the increased load on the HSS and the reduced performance of the BSF.
In the fifth implementation (see Figures 16 to 18), a GUSS is generated "on-the-fly" from a GUSS template, or by other means, by a proprietary extension of the HSS functionality. The fifth implementation has the potential advantage that the GUSS scripting capability in the HSS might be useful for other (unforeseen) purposes. Disadvantages include the need for a proprietary extension to the HSS.

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 service provider, wherein the service provider includes a resource and wherein the service provider requires a user requesting access to the resource to be authenticated using generic bootstrapping architecture, the service provider comprising:
   a first input for receiving a request to access the resource from a first user device, wherein the first user device is referred to the resource by a second service provider; and
   means for obtaining one or more registered IP multimedia public identities for the first user device as registered for an IMS communication session between the first user device and a second user device.

2. A service provider as claimed in claim 1, wherein said one or more registered IP multimedia public identities are received at the service provider in a request received at said first input from the first user device requesting access to the resource.

3. A service provider as claimed in claim 2, wherein said service provider obtains a list of all IP multimedia public identities for said first user device and compares the IP multimedia identities included in the list with the one or more registered IP multimedia identities included in the request received from the first user device.

4. A service provider as claimed in claim 2 or claim 3, wherein the one or more registered IP multimedia public identities for the first user device are stored at the first user device during an IMS registration process.
5. A service provider as claimed in claim 1, wherein said one or more registered IP multimedia public identities are obtained by the service provider.

6. A service provider as claimed in claim 5, wherein the service provider is adapted to contact a home subscriber service for the first user device in order to obtain said one or more registered IP multimedia public identities.

7. A service provider as claimed in claim 5, wherein the service provider is adapted to contact a bootstrapping server function for the first user device in order to obtain said one or more registered IP multimedia public identities.

8. A service provider as claimed in claim 5, wherein a home subscriber server for the first user device generates a list of registered IP multimedia public identities for the first user device during the GBA bootstrapping procedure, and wherein said service provider is adapted to obtain said one or more IP multimedia public identities obtained by requesting said list.

9. A method comprising:
   receiving a request at a first service provider for access to a resource provided by the first service provider, wherein the request is received from a first user device and wherein the first user device is referred to the resource by a second service provider; and
   authenticating the first user device at the first service provider in accordance with generic bootstrapping architecture protocol, wherein authenticating the first user device includes obtaining one or more registered IP multimedia public identities for the first user device as registered
for an IMS communication session between the first user device and a second user device.

10. A method as claimed in claim 9, wherein said one or more registered IP multimedia public identities are received at the first service provider in a request sent by the first user device to the first service provider requesting access to the resource.

11. A method as claimed in claim 10, wherein said first service provider obtains a list of all IP multimedia public identities for said first user device and compares the identities included in the list with the one or more registered IP multimedia identities included in the request received from the first user device.

12. A method as claimed in claim 10 or claim 11, wherein the one or more registered IP multimedia public identities for the first user device are stored at the first user device during an IMS registration process.

13. A method as claimed in claim 9, wherein said one or more registered IP multimedia public identities are obtained by the first service provider.

14. A method as claimed in claim 13, wherein the first service provider contacts a home subscriber server for the first user device in order to obtain said one or more registered IP multimedia public identities.

15. A method as claimed in claim 13, wherein the first service provider contacts a bootstrapping server function for the first user device in order to obtain said one or more registered IP multimedia public identities.
16. A method as claimed in claim 13, wherein a home subscriber server for the first user device generates a list of registered IP multimedia public identities for the first user device during the GBA bootstrapping procedure, and wherein said one or more registered IP multimedia public identities obtained by the first service provider are obtained by requesting said dynamic list.

17. A method as claimed in any one of claims 9 to 16, wherein the second user device instructs the first user device to contact the service provider by means of an SIP REFER message.

18. A user device comprising:
   a first input for receiving a referral message instructing the user device to access a resource at a first service provider, wherein the first service provider is a network application function that requires the first user device to be authenticated using a Generic Bootstrapping Architecture protocol; and
   a first output for sending a request for access to the resource at the first service provider, wherein the user device is adapted to provide one or more registered IP multimedia public identities for the first user device as registered for an IMS communication session between the first user device and a second user device to the first service provider.

19. A system comprising a first user device, a first service provider and a second service provider, wherein:
   the first user device is adapted to communicate with a second user device via the second service provider, wherein the second service provider is an IMS application server;
the second service provider is adapted to instruct
the first user device to contact the first service provider,
wherein the first service provider is a network application
function that requires the first user device to be authenti-
cated using Generic Bootstrapping Architecture; and

the first service provider receives one or more IP
multimedia public identities for the first user device as
registered for the IMS communication between the first and
second user devices.

20. A computer program product comprising:

means for receiving a request at a first service
provider for access to a resource provided by the first ser-
vice provider, wherein the request is received from a first
user device and wherein the first user device is referred to
the resource by a second service provider; and

means for authenticating the first user device at
the first service provider in accordance with generic boot-
strapping architecture protocol, wherein authenticating the
first user device includes obtaining one or more registered
IP multimedia public identities for the first user device as
registered for an IMS communication session between the first
user device and a second user device.
Fig. 2

Sequence Diagram:

1. UE
2. BSF
3. HSS
4. NAF

flows:
- Unauthorised: 2 -> 4
- Unauthorised: 4 -> 6
- Authorized: 4 -> 11
- Authorized: 11 -> 12
- Authorised: 6 -> 12
- Authorised: 12 -> 14
- Request: 12 -> 20
- Generate B-TID: 20 -> 22
- Application Request: 22 -> 23
- Application Response: 23 -> 28
- Application Response: 28 -> 32

Steps:
1. UE sends an Unauthorised request to BSF.
2. BSF sends the request to HSS.
3. HSS authorises the request and sends the Authorised message to BSF.
4. BSF generates the B-TID and sends the Application Request to NAF.
5. NAF sends the Application Response to BSF.
6. BSF sends the Application Response to UE.

Fig. 2
Fig. 11

1. IMS initiation
2. HTTP Referral
3. GBA Authentication
4. SP obtains IMPUs
Fig. 14

152 Request 2 to SP

154 Authentication

156 Obtain IMPUs

158 Check IMPUs

159 Grant Access

Fig. 15
IMS initiation

HTTP Referral

GBA Authentication

HSS obtains IMPUs

Fig. 16
**INTERNATIONAL SEARCH REPORT**

International application No  

PCT/EP2010/055105

A. CLASSIFICATION OF SUBJECT MATTER  

INV. H04L29/06

ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L

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

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

EPO-Internal, WPI Data

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>
<tbody>
<tr>
<td>X</td>
<td>Friese L et al.: &quot;Bringing IMS and Internet Identity&quot;, Liberty Alliance Project, SIG Tel ecommunications, 1 December 2009 (2009-12-01), pages 1-44, XP002629829, Retri eved from the Internet: URL: <a href="https://www.surfgoopen.nl/si">https://www.surfgoopen.nl/si</a> tes/dmne twork/Shared%20Documents/2009-12-03-i_dm-Li berty-Al liance-WP-Bri ngngMS_A ndInternetI dentity_V1.0a.pdf [retrieved on 2011-03-24] page 9, line 321 - line 343; figure 2</td>
<td>1-20</td>
</tr>
</tbody>
</table>

Further documents are listed in the continuation of Box C.  

See patent family annex.

A 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 not later than the priority date claimed

Date of the actual completion of the international search  

25 March 2011

Date of mailing of the international search report  

09/05/2011

Name and mailing address of the ISA/  

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040;  
Fax: (+31-70) 340-3016

Authorized officer  

Tabery, Peter
<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>
<tbody>
<tr>
<td>Patent document cited in search report</td>
<td>Publication date</td>
<td>Patent family member(s)</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-----------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>WO 2010041347 A1</td>
<td>15-04-2010</td>
<td>NONE</td>
</tr>
<tr>
<td>US 2008120705 A1</td>
<td>22-05-2008</td>
<td>NONE</td>
</tr>
</tbody>
</table>