A system for mutually authenticating links between people, entities, objects and devices comprises at least two wireless devices 116, 110 capable of short range communication (Bluetooth, WiFi, infrared etc), each having an associated server database 118, 112 storing unique identifiers and between which authorisation data is transmitted. A first wireless device 116 searches for proximal wireless devices and graphically displays them. On selection of a displayed device 110, an IP address of a server 112 is determined from which access authorisation to the device 110 is requested. The server 112 authenticates the user of the first device 116 by requesting verification from the displayed device 110 or from a server 118 associated with the first device. On verification the user of the first device may access personal data associated with the user of the device 110, control the device 110 or access a service offered by the server 112. Embodiments also relate to wirelessly unlocking vehicles, billboard advertising, cash point machines, currency transfers, voting and banking.
<table>
<thead>
<tr>
<th>Unique DB ID</th>
<th>Short Description</th>
<th>Data</th>
<th>Linked/Authorised</th>
</tr>
</thead>
<tbody>
<tr>
<td>501</td>
<td>User's Unique ID</td>
<td>&lt;PUBLIC&gt;ABC, &lt;PRIVATE&gt;123, &lt;BOND&gt;</td>
<td></td>
</tr>
<tr>
<td>502</td>
<td>Mobile Phone 1</td>
<td>Mobile Phone's Current IP address</td>
<td></td>
</tr>
<tr>
<td>503</td>
<td>Car</td>
<td>Car's Current IP address</td>
<td>123123123123501</td>
</tr>
<tr>
<td>504</td>
<td>Home</td>
<td>Home's System Current IP address</td>
<td>123123123123501</td>
</tr>
<tr>
<td>510</td>
<td>Name</td>
<td>John Smith</td>
<td></td>
</tr>
<tr>
<td>511</td>
<td>Phone Number</td>
<td>+44567890099</td>
<td></td>
</tr>
<tr>
<td>512</td>
<td>Email</td>
<td><a href="mailto:JohnSmith@domain.com">JohnSmith@domain.com</a></td>
<td></td>
</tr>
<tr>
<td>555</td>
<td>Private Link</td>
<td>1231231231235511, 1231231231235512,</td>
<td></td>
</tr>
<tr>
<td>556</td>
<td>Business Link</td>
<td>1001001001008021</td>
<td></td>
</tr>
<tr>
<td>561</td>
<td>Big Business Ltd</td>
<td>1001001001008021</td>
<td></td>
</tr>
<tr>
<td>562</td>
<td>Joe Blog</td>
<td>0909090909094021</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 5**

<table>
<thead>
<tr>
<th>Unique DB ID</th>
<th>Short Description</th>
<th>Data</th>
<th>Linked/Authorised</th>
</tr>
</thead>
<tbody>
<tr>
<td>601</td>
<td>User's Unique ID</td>
<td>PUBLIC ABC,PRIVATE 123, &lt;BOND&gt;</td>
<td></td>
</tr>
<tr>
<td>602</td>
<td>Mobile Phone 1</td>
<td>Mobile Phone's Current IP address</td>
<td></td>
</tr>
<tr>
<td>603</td>
<td>Car</td>
<td>Car's Current IP address</td>
<td></td>
</tr>
<tr>
<td>604</td>
<td>Home</td>
<td>Home's System Current IP address</td>
<td></td>
</tr>
<tr>
<td>610</td>
<td>Name</td>
<td>Joe Blogs</td>
<td></td>
</tr>
<tr>
<td>611</td>
<td>Phone Number</td>
<td>+44567890099</td>
<td></td>
</tr>
<tr>
<td>612</td>
<td>Email</td>
<td><a href="mailto:JoeBlogs@domain.com">JoeBlogs@domain.com</a></td>
<td></td>
</tr>
<tr>
<td>655</td>
<td>Private Link</td>
<td>0909090909094011, 0909090909094012,</td>
<td></td>
</tr>
<tr>
<td>661</td>
<td>John Smith Private</td>
<td>1231231231235521</td>
<td></td>
</tr>
<tr>
<td>662</td>
<td>John Smith Bus</td>
<td>1001001001008021</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 6**

<table>
<thead>
<tr>
<th>Unique DB ID</th>
<th>Short Description</th>
<th>Data</th>
<th>Linked/Authorised</th>
</tr>
</thead>
<tbody>
<tr>
<td>701</td>
<td>User's Unique ID</td>
<td>PUBLIC EPD,PRIVATE 456</td>
<td></td>
</tr>
<tr>
<td>702</td>
<td>Desktop 12</td>
<td>Desktop 12</td>
<td></td>
</tr>
<tr>
<td>703</td>
<td>Company Car 15</td>
<td>Car's 15 Current IP address</td>
<td></td>
</tr>
<tr>
<td>704</td>
<td>Office Entrance</td>
<td>Office Entrance Current IP address</td>
<td></td>
</tr>
<tr>
<td>705</td>
<td>Business Parking</td>
<td>Business Parking Current IP address</td>
<td></td>
</tr>
<tr>
<td>706</td>
<td>Security Desktop 8</td>
<td>Security Desktop Current IP address, &lt;BOND&gt;</td>
<td></td>
</tr>
<tr>
<td>710</td>
<td>Business Name</td>
<td>Big Business Ltd</td>
<td></td>
</tr>
<tr>
<td>711</td>
<td>Main Phone No</td>
<td>+44555444431</td>
<td></td>
</tr>
<tr>
<td>712</td>
<td>Address</td>
<td>100 Green Avenue, London, UKW1 1AA</td>
<td></td>
</tr>
<tr>
<td>720</td>
<td>John Smith Business</td>
<td>1001001001008011, 1001001001008012,</td>
<td></td>
</tr>
<tr>
<td>777</td>
<td>John Smith Private</td>
<td>1231231231235521</td>
<td></td>
</tr>
<tr>
<td>781</td>
<td>John Smith Access</td>
<td>1001001001008001, 1001001001008002,</td>
<td></td>
</tr>
<tr>
<td>791</td>
<td>Joe Blog</td>
<td>0909090909094021</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 7**
A method and system for the creation, management and authentication of links between people, entities, objects and devices

Background

The present invention relates to a communication system which simplifies the development, management and authentication of relationships between people, organisations, objects and machines.

Currently, mobile phone users have the option of sending a business card using short range communication including but not limited to Bluetooth (RTM), infrared, or long range communication including but not limited to GPRS or UMTS. However because the information is sent as a business card to a recipient he or she is not able to confirm authenticity of the data received nor manage relationships and ownership of the contact data or bonding between people, organisations, objects and machines in order to get access to information or physical devices in real time.

Additionally there are problems such as that users have to remember a multitude of passwords to access for example websites, the impossibility of verifying the true identity of a user for authorising access to public transport, or an international border, or for exchanging currency. As it will become apparent, the present invention has many uses and many more will be found beyond those described in this document.

In the closest documented systems, there are suggestions of static contact information from mobile phones to be wirelessly synchronised to a single remote server, which requires people or businesses to put their employee’s personal data, and their business relationships into the hands of the service provider controlling the centralised system.

In another example, existing mobile phone contact information is uploaded to a remote server and then synchronised between remote servers. This has serious practical limitations as the data must initially be manually entered by, and exchanged between different users.

In all cases, these proposed systems have not satisfactorily recognised the difference between machines, objects, people and organisations, nor recognised them as separate entities.
Object of the invention

It is an object of this invention to provide a system and method for at least one of forming, managing and authenticating relationships between at least two of people, entities and machines.

Summary of the Invention

In order to provide authenticated access for a person, to personal data of his contacts, to control a device, or to access an electronic service, the user is provided with a server. He registers his mobile device to his server and his data, where after the server authenticates the mobile device as representing him.

When he attempts to use his mobile server to access such data, device or service, provided by a remote system, that remote system will record a unique identifier of his mobile device. Means are provided for permitting the remote system to determine the user’s authentication server. These means are either by making a request to a pointer server, or by encoding this information in data sent from the user’s mobile device to the remote system (E.g. Encoded within the unique identifier), where a short range communication (Bluetooth, WiFi or Infrared) is used at least in one step of the process.

The remote system requests confirmation from the user’s authentication server that the user is indeed using the uniquely identified mobile device, and further that the mobile device is being used to access the remote system. On confirmation by return, the user via his mobile device is authenticated, and is authorised to access the desired resource.
Description

After registering the necessary information for his identity and logging onto the system using a mobile device, a user may search for nearby enabled devices, determine at least the names of the users of those devices, and select a link to a set of information that other users may have access to. Once given access, those users will continue to have access in the future (if access was not time limited). Additionally, when the devices are then separated by physical distance, the internet may be used to access this information. Further, a user may gain access to authorised resources using alternate devices, for example if he lost his mobile phone. After the creation of such a link between the user and an entity (a resource or another user), he might be provided with access to additional resources linked to that entity. These resources may be accessed using short range communication (for example a business computer, public transport, a cash point, a car or house), which may be achieved via connections to the internet to the user’s system (or their service provider system which ultimately connects to the user’s system). Additionally he can access particular information via an internet connection even where short range communication is not available. Although the process involves particular data including but not limited to a Unique ID exchanged between devices, the links can be formed between the users of the system (more specifically between any of users/entities/devices and other users/entities/devices).

The separation of server functionality into pointer servers and authentication servers may be used only to bypass the need to exchange an IP address of an authentication server and at the same time avoid the need for users to store personal data on a server that they do not have control over (i.e. the data can be on a company’s server, and pointer servers indicate that the data for certain devices and users is on that company server – but a pointer server provider company need not have access to the data).

A general principle is the provision of two-way authentication for access, which is made possible by providing each device owner with a server (whether an individual or a company), and avoiding the need to place (personal or company) data into a 3rd party centralised server.
Different methods are possible for one device to determine the location of the authentication server of the other device as will be detailed.

Generally the system uses unique identifiers, including but not limited to the serial numbers of the communications modules of the devices. Additionally there is the feature of verification of identity, to prevent such identifiers being spoofed. Verification is generally performed between two authentication servers

Explanations of terms used to define system elements:

The term “Entity” is used to identify system’s: users, organisations, companies, governments, institutions, associations, establishments, societies, bodies, objects, devices etc. It has a distinct, separate existence, though it need not be a physical existence.

The term “Object” generally corresponds directly to a contiguous block of computer memory of a specific size at a specific location (although it would be possible to use non-contiguous blocks, i.e. virtual blocks). This could be a file of any type including but not limited to text, picture, sound, video or spatial coordinates, HTML, XML, Binary, formatted as a web page, driver for device, program code compiled or not etc.

Throughout this document, references to an object being a collection of data identified by a unique ID on a user’s authentication server database. For example the row next to number 555 in figure 5.

Additionally an object could be an individual unit of run-time data storage that is used as the basic building block of programs. Opposed to a traditional view of a program seen as a collection of functions, or simply as a list of instructions to a computer these objects act upon each other. Objects are capable of receiving messages, processing data, and sending messages to other objects. Each object can be viewed as an independent virtual machine with a distinct role or responsibility.

The term “Unique ID” or Unique Identification Code refers to a data set which is unique for a particular system. It could be generated for general use, or could be dynamically unique, being a specific unique ID for a particular server or database. Alternatively it could be a number used to electronically identify a specific module participating in the system, built-in to the device (for
example BD_ADDR, BSSID, SSID, manually set device name of a Bluetooth (RTM) or WiFi module). The unique ID will directly or indirectly identify an Entity (which could be a User, an organisation, an object, a Machine or an electronic module built in to the machine). Directly Unique Entity ID will resolve to identity of a user, while indirectly a Unique ID (usually identifying device) will resolve Unique Entity ID and/or identify a user.

In an example system and method below there is reference to a Unique ID as a combination of a 32 bit IP address for a particular authentication server with a unique number for the particular user and also a number of a particular object for that particular user. For example in figure 5 the database row next to the number 555 represents an object identifying the “Private Link” of a user called “John Smith” and is made of the IP address of John’s authentication server (123.123.123.123) the unique number for John (55) and the unique number for John’s object called private link (21), thus the final Unique ID identifying John’s private link is 1231231231235521.

Additionally the system could be designed so that the unique number (or part of it), is periodically changed and updated to the mobile phone device, desktop and all authentication servers and pointer servers connected to it so as to provide greater privacy to the user of the system from malicious monitoring of the Unique ID.

Furthermore, where 128bit IP addresses are widely available, it will be possible to have a system where just the IP address will uniquely identify each entity. Indeed each unique IP address could represent an object/row in a local database, which would become essentially a unique link between two entities, and thus it would not be necessary to append to an IP address a unique number for a particular user or a number of an object. This will be possible due to the fact that an IPv6 address is divided in two logical parts: a 64-bit (sub-) network prefix, and 64-bit host part. The 64-bit host part could be used for the definition of the exact user and/or object and server as a link between two entities and/or devices. Moreover each link could dynamically change its 64-bit host part over time thus increasing a user’s privacy, and potentially contributing to the security of the system. Additionally 64-bit host part of a system device could be changed so it represents an IP address of own authentication server, thus any connected device and authentication
server connected to it would be able to contact responsible authentication server for any device by just reading 128 bit IP address version 6 of that device. This could simplify system architecture and make system function much faster.

Another type of number used could be a Universally Unique Identifier (UUID) which is an identifier standard used in software construction, standardized by the Open Software Foundation (OSF) as part of the Distributed Computing Environment (DCE). It has about $3.4 \times 10^{38}$ combinations which mean that 1 trillion UUIDs have to be created every nanosecond for 10 billion years to exhaust the number of UUIDs.

Yet another type of unique ID could be a phone number as it is internationally unique and could be used for initialisation of the system involving but not limited to SMS, MMS, WAP push etc.

Note that if the system or a part of the system is using a Unique ID without included an authentication server IP address, and no pointer server is involved, then a Unique ID must be transferred directly to the authentication server and/or entity/device at least initially using SMS, MMS, email, voice, manual input, or by other means and be linked in at least one database to that particular entity and/or device.

The term “Bonding” (“Linking”) is a method for building relationships between Entities (including but not limited to machines, users, organisations, objects or institutions), which will be explained in more detail below.

Typically an ordinary person (The Entity) will use their mobile phone device with a short range communication module for bonding with friends, business colleagues, banks, governments, cars, houses etc. while an Entity such as an institution might prefer to use a desktop with a short range communication module.

After bonding is completed, the mobile phone device user might get access to other machines or information linked to that entity utilising short range communication, (for example a business computer, public transport, a cash point, car or house etc), which may all be connected via the internet to the user’s system (or their system service provider which ultimately connects to the user’s own system)
The term "Mobile Phone Device" relates to: A display, a keypad, Read-only Memory (ROM), Random Access Memory (RAM), a long range radio transmitter and receiver for systems (which may include Global System for Mobile Communications (GSM) and its subset General Packet Radio Service (GPRS) or Universal Mobile Telecommunications System (UMTS)), a short range communication module (E.g. Bluetooth (RTM), WiFi and Infrared), a Central Processing Unit (CPU), Speaker, Microphone, Battery, Operating System (OS), software drivers and applications installed on top of the OS necessary for the functioning of the mobile phone device. Mobile phones which permit access to the OS and permit third party software to be installed are known as 'smartphones', but it is not necessary for the mobile phone device to have this functionality for utilising the present invention.

A typical graphical symbol used in this document to represent a mobile phone device is number (110) in figure 1a.

The purpose of the mobile phone device within the system is to build, manage and view links between machines, people and entities using its Unique ID.

The term mobile phone device includes mobile phones, Personal Digital Organisers, PC's, Laptops, tablet computers and any terminals with Internet connectivity and/or short range communication.

A Desktop Computer with Internet and Bluetooth (RTM) module, for the purposes of this document consists of: a display, a keyboard, a pointing device, Read-only Memory (ROM), Random Access Memory (RAM), a Hard Disk with a database, a network card or modem (wireless or not) for accessing the Internet, a short range communication module (Bluetooth (RTM), WiFi, Infrared), a Central Processing Unit (CPU), an Operating System (OS), software drivers and applications installed on top of the OS necessary for the functioning of the desktop computer.

An example of a graphical symbol depicting a desktop computer is number (323) in figure 3.

The purpose of such desktop machines is to build, manage and view links between devices and entities using a Unique ID.
The term desktop computer could include mobile phones, Personal Digital Organisers, PC's, Laptops, tablet computers and any terminals with Internet connectivity and/or short range communication.

The term "Pointer Server" consists of: Read-only Memory (ROM), Random Access Memory (RAM), a Hard Disk with a database, a network card or a modem (wireless or not) for accessing the Internet, a Central Processing Unit (CPU), a power supply, an Operating System (OS), software drivers and applications installed on top of the OS necessary for the functioning of the pointer server.

The purpose of the Pointer Server in this document is to accept a Unique ID sent typically via a short range communication link between two devices, and then transmitted to the Pointer Server via the Internet, and particularly to relate that Unique ID to a location (including but not limited to an IP address or a Domain Name) of the authentication server which is responsible for authentication of that Unique ID.

The separation of the pointer server and the authentication server means that the Pointer Server does not need to be updated each time the user bonds with an Entity, and allows a situation where no other data related to the entity is stored remotely where it might be compromised.

The pointer Server's IP address is publicly known by all participants in the system, and every authentication server should store that IP address for use.

An example of a graphical symbol depicting a Pointer Server is next to the number (114) in figure 1a.

An Authentication Server consists of: Read-only Memory (ROM), Random Access Memory (RAM), a hard disk with a database, a network card or modem (wireless or not) for accessing the Internet, a Central Processing Unit (CPU), a power supply, an Operating System (OS), software drivers, applications and databases installed on top of the OS necessary for the functioning of the authentication server.

The purpose of the authentication server is to relate a Unique ID to a particular entity and/or device. Furthermore an authentication server is used as a relationship manager between the entity and other entities linked to it.
Typically all user devices including a mobile phone device or a desktop will be connected directly or indirectly to at least one authentication server.

Additionally the authentication server for a particular user could be a mobile phone device itself, holding a database with Unique IDs thereon. A disadvantage of such a system is that when a device is not connected to the network, (E.g. due to poor network coverage, low battery or due to loss or theft) it will be absent from the system and thus unavailable for other connected entities connecting in the background (for example updating or requesting information).

Some of these issues could be resolved by adding a back-up system but device availability would remain low and data transfer to and from the device would remain high and possibly cost prohibitive in the short to medium term future.

An example of a graphical symbol used to represent an authentication server is number (118) in figure 1a (with the exception that (312) is used as a Storage Server in one example).

An **Entity Name Server** consists of: Read-only Memory (ROM), Random Access Memory (RAM), a hard disk with a database, a network card or modem (wireless or not) for accessing the Internet, a Central Processing Unit (CPU), a power supply, an Operating System (OS), software drivers and applications and databases installed on top of the OS necessary for the functioning of the Entity Name Server.

The purpose of the Entity Name Server is to enable users to login remotely using any mobile phone device (or wireless machine or desktop) and gain access to resources they previously had registered on their authentication server. The Entity Name Server additionally enables single password login through a website, and for people (or other Entities) to be located by searching a database held by the Entity Name Server.

All information about users (and other entities) comes to an Entity Name Server from respective Authentication Servers and is available to all those wishing to perform world-wide searches.

Another purpose of Entity Name Server is to link an old Unique Entity ID to the new one (with permission from the owner) if for example
authentication server IP address has changed or simply because user’s data has moved to another authentication server.

An Entity Name Server is optional to the system and an example representation is number (121) in figure 1a.

A Storage Server consists of: Read-only Memory (ROM), Random Access Memory (RAM), a hard disk with a database, a network card or modem (wireless or not) for accessing the Internet, a Central Processing Unit (CPU), a power supply, an Operating System (OS), software drivers and applications and databases installed on top of the OS necessary for the functioning of the Storage Server.

The purpose of a storage server is to enable sharing of other data from a remote location, for example pictures, music, video, etc.

An Entity’s Links are data sets stored on the authentication server and generally may be synchronised to a mobile phone device and/or a desktop computer. These data sets consist of data identifying the user (or other entity) or device.

An entity link can have data/information attached to it and can be a pointer to another link.

Every Entity has two types of links: An “Own Link” which identifies that particular Entity, and is usually exchangeable, and an “Other Entity link” which is typically not exchangeable. The purpose of the “Other Entity Link” is to have another entity’s contact details always available and up to date, or to enable access to their resources (including but not limited to a website, car, bank account etc).

Every link between different entities can have a different status attached to it, such as:

- Exchangeable’ – where the link is always exchangeable between different parties, without requiring further authorisation.
- Exchangeable with permission’ – where the link is exchangeable between different parties but does require further authorisation from link owner for this
- Exchangeable with introduction’ – where the link is exchangeable between different parties, but only if the entity to be introduced comes from an
already trusted/linked source and therefore does not require further authorisation from the link owner.

‘Exchangeable with permission and introduction’ – where the link is exchangeable between different parties but only if the entity to be introduced, comes from an already trusted/linked source, and there is further authorisation from the link owner.

‘Non-exchangeable’ – where the link is never exchangeable between different parties.

The "Own Link" consists of the links to a user’s own telephone numbers and personal data (E.g. pictures, addresses, videos etc) such as "Private Link" in Figure (555). Alternatively the data in the "Own Link" could be pointing to a remote authentication server, if for example a user’s employer wanted to keep their own data and be able to change it as necessary. This is achieved by adding a "Business Link" row (556) in figure 5 to the user’s authentication server database issued by a user’s Business Authentication Server as represented by number (777) in figure 7. The user typically has the option to exchange these parts of the link with other entities (including but not limited to people, organisations, objects or machines).

The "Other Entity Link" consists of data which relates to other people and entities. It is not editable by a recipient/viewer user, only by the originator/owner and is represented by the database row next to the number (562) in figure 5. The main purpose of this link is to have up to date, rich, contact details of other people and/or entities (E.g. name, telephone number, pictures, video, business contact data etc).

Both parts, the "Own Link" and the "Other Entity Link" are stored locally by the mobile phone device and the respective authentication server, and checked as necessary with the originator for their current validity. Otherwise every originator of the information authentication server will notify all users connected to its link as soon as there is any change of information. In the case that the data is not stored temporarily locally there might be problems with slow connections preventing the retrieval of information as needed.

Differentiation between the “own link” and the “other entity link” could be achieved at the software level or by constructing a database in such way that a range of object numbers in the Unique ID identifies one from the other.
Other methods might include inserting a tag in the data field or adding a new column to the database.

A "Device" is any electrical device for handling a particular type of information and performing related tasks. Every device should have at least one Central Processing Unit (CPU) or Microcontroller or other active electronic component for processing and storing information.

For a better understanding of the invention and to show how the same may be carried into effect, reference will now be made, by way of example, to the accompanying drawings, in which:

Figure 1a shows a schematic of a system according to an embodiment of the present invention, using external authentication server architecture and a pointer server,

Figure 1b shows a schematic of the system according to another embodiment, without external authentication server and/or a pointer server,

Figure 1c shows a schematic of the system according to another embodiment, where only one device has access to the internet,

Figure 2 shows a schematic of how a system from figure 1 can be expanded vertically i.e how more pointer servers could be added for faster and simpler communication,

Figure 3 shows a schematic applicable to many different embodiments of the system,

Figure 4 is a graphical representation of devices and authentication servers linked to a single user or entity,

Figure 5 is a simple representation of a database structure and underlying data used by first user,

Figure 6 is a simple representation of database structure and underlying data used by second user, and

Figure 7 is a simple representation of database structure and underlying data used by first user employer.

In the following description, various aspects of the present invention will be described. For purposes of explanation, specific configurations and details
are set forth in order to provide a thorough understanding of the present invention. However, it will also be apparent to one skilled in the art that the present invention may be practiced without the specific details. Furthermore, well known features may be omitted or simplified in order not to obscure the present invention.

According to one aspect of the present invention there is provided a networking system for facilitating the establishment of a relationship between a user and a remote device or the user of a remote device, - the networking system comprising: A first system having: A first device, being a wireless mobile Internet connected device being identifiable by a unique identification code, controlled by a user, adapted to search for proximal wireless devices in response to a user's input via user interface means, adapted to graphically represent identified devices, and adapted to accept a selection of one thereof by the user, A first server related to the first device, provided with authentication data relating to the first device, A second system having, A second device being a wireless device identifiable by a unique identification code, and A second server related to the second device, provided with authentication data relating to the second device, Means for providing an IP address of the second server to the first system, - the networking system specifically adapted such that; On user selection to the first device, of a graphical representation of the second device, the first system is specifically adapted to: determine the IP address of the second server, send a data request to the second server, encoding a request for at least one of the first device and its user, to be henceforth authorised to access the second system, On receipt of such a data request, the second server is adapted to authenticate the user of the first device by at least one of: Requesting verification from a user of the second device, via user interface means of the second device, where access to personal data relating to the user of the second device is requested, or Making a data request to the first server, encoding a request for verification that a user of the first device is seeking such authorisation, where access to a service offered by the second system is requested, On verification, the user of the first device may use the first device to either: access personal data associated with user of the second device,
control the second device, or access a service offered by the second server. In the case that the user of the first device instead chooses to use an alternate device, the second server is specifically adapted to: accept remote logon by the user of the first device, with personal user data via the first device, and update at least one of the second server and the alternate first device with authentication data, such that the user may continue to access such personal data, service or control of the second system, via the chosen alternate first device.

According an embodiment of the present invention there is provided a method for facilitating exchange/transfer of a unique identification code between devices with the purpose of relating/linking an entity related to a device and/or an entity related to a device, the system comprising: A first device, a wireless mobile internet connected device being: Identifiable by a unique identification code related to an entity, Controlled (automatic?) by an entity, Adaptable to exchange with other proximal wireless devices the unique identification code using short range communication, Adaptable to exchange with other proximal wireless devices the unique identification code using short range communication in response to a user of a wireless device via user interface means, Adaptable to check identity of entity behind unique identification code using internet connection, Adaptable to graphically represent identified entities, and adapted to accept selection of identified entities, Adapted to make selection which data sets/links should be related to the selected entity, Adapted to write selection in own database, Adapted to transfer this information to the entity selected, A first server/database connecting to the first device, provided with authentication data relating to the first device, A second device a wireless device identifiable by a unique identification code, and A second server/database connecting to the second device, provided with authentication data relating to the second device, Means for providing an IP address of the second server to the system/server, The system specifically adapted such that; On user activation to the first device, short range communication to collect unique identification code of the second device, Determine IP address of the second server related to the second device, encoding a request for the second server to provide identity of the entity related to the second device, Receive and display identity of the
entity related to second device. On user selection to the first device, of
graphical representation of identity of the entity related to the second device,
the first system is specifically adapted to: Make selection of specific
authorisation to access the first device entity information or resources, Encode
and transfer such selection to the first server to be written to the first server
database and to forward authorisation to the second server related to the
second device, After verification of identity, the entity using the first device
may use the first device to either: Access authorised personal data associated
with second entity, Control second device, Access the service offered by the
second entity.

According to another embodiment of the present invention there is
provided a system for facilitating transfer of a unique identification (ID) code
between devices with said purpose of linking an entity related to a device and
a device and/or an entity related to a device. Said system of at least two
separate devices comprising: A first device, a wireless terminal with at least
one internet connection and at least one short range communication
connection, A second device, a wireless terminal with at least one short range
communication connection, At least two separate databases, each containing
at least one said unique identification (ID) code.

According to another embodiment of the present invention there is
provided a method for facilitating the establishment of authorisation of a user
for accessing a service, having the steps of: Registering the user to the user’s
mobile device, Registering the mobile device to the user's data server,
Identifying a unique ID of a remote device wirelessly using the mobile device,
Establishing the IP address of an authentication server associated with the
unique ID or remote device, Sending a request to that authentication server,
encoding a unique ID of the user's mobile device therein, the request being to
authorise the user to access a service, Accepting a data request to the server,
the request being to confirm that the mobile device is being used for access to
the authentication server, Verifying this by communication between the server
and the user's mobile device, Sending the requested confirmation by return,
and, Accessing the service using the user’s mobile device.

According to another embodiment of the present invention there is
provided a device specifically for performing the role of one of mobile device,
remote wireless device, and authentication server, being specifically adapted for such role.

According to another embodiment of the present invention there is provided a computer program specifically adapted for one of the mobile device, remote wireless device, and authentication server in the previous embodiment, for controlling such hardware for use in the system described in another embodiment.

Further embodiments are provided by the selection of any combination of features hereinbefore set out.

Further Description

Once software, (which may be downloaded after purchase of a mobile phone device or loaded during the mobile phone device’s manufacturing process), is installed and run for the first time on the mobile phone device (110) as in figure 1a, it will register with an authentication server (112) using the internet connection (111). Typically the Internet connection would be via GPRS or UMTS but it could be any other type of data transfer between a mobile phone device with a short range radio connection (115) including but not limited to Bluetooth (RTM) or WiFi or Infrared and the authentication server.

If this is the first time that the user is using the service, the authentication server will register the new user or entity by creating a new account in its database using some unique identification. The mobile phone device unique ID could be randomly generated by the system or predetermined by the hardware or software of the authentication server (112) or the mobile phone device (110). In one preferred embodiment, the mobile phone device will provide an electronically engraved BD_ADDR number. This is a 42 bit IEEE Bluetooth device address unique to each Bluetooth (RTM) module (or BSSID for WiFi) which will be registered in the database.

There are two main methods for exchange of links as follows.

The first method is where an IP address of an authentication server is exchanged and available, possibly using a User ID, in which case there is no need for a Pointer Server (114) in Figure 1 and (114, 201) in Figure 2.
The first method could be achieved by changing the Bluetooth (RTM) name of a mobile phone device to contain an authentication server IP address and/or a User/Device unique ID and/or an Object ID as necessary. For example if John Smith’s authentication server’s IP address is 123.123.123.123 and his User ID for this particular authentication server is 55 and Object ID is 01 then his unique ID would be 1231231231235501, which could be embedded in John’s mobile phone device name. His new mobile phone device name might look like “John@1231231231235501”, from which any other device could read the location of John’s authentication server as well as the Unique User/Device ID and Object ID, if necessary. Note that the word ‘John’ before the “@” symbol and the Unique ID number are not necessary for the system to function. Yet another way to exchange the Unique ID with or without including an IP of an authentication server would be to use the OBEX communication protocol over the short range communication link, whereby all necessary information could be passed, including but not limited to an IP address of an authentication server and/or a User ID and/or an Object ID.

A second method is where only the Unique ID is known but not an authentication server’s IP address responsible for that particular Unique ID (in this example the BD_ADDR of a Bluetooth (RTM) module). This problem is overcome by using a Pointer Server, which relates any Unique ID to the IP address of its authentication server. The authentication server will further resolve data relating to the entity and any links it has towards other entities or devices.

However, in an example that follows, there will be an exchange of a Unique Device ID including but not limited to BD_ADDR without an authentication server IP address being exchanged in a short range communication, but instead using a Pointer Server. Note that any other short range wireless radio identification could be used including but not limited to a WiFi device name or a Service Set Identifier (SSID) identification number (access point number) (BSSID) or in some cases the MAC address of a network card or other hardware specific unique numbers or even user’s phone number. Alternatively the unique number could be a static or dynamic number generated by the system on the server or client side.
The system will generate a unique entity ID and related data sets for the user on the authentication server. So even if a mobile phone device changes or if it changes data connection, the entity's unique identification and links to other users and entities can be maintained. During the registration process a user can supply any other information (e.g. their name, telephone number, email, picture, video etc) which is then directly linked to them in the database as well as a multitude of different possible definitions of links (e.g. Private, Business Hobby, Charity, etc). These links could be added not just by a user but by other machines or users or entities as will be discussed later in the description. Additionally a user will supply two passwords which will be stored in the database, including a first private password, for him to log onto the system so he can later add devices, change his data or manage links. Additionally the user will supply a second public password, which may be used as verification of his identity, for example to manually access a website which would automatically be populated with his contacts (for example if it was an email provider) or other data (for example for a social networking website). Note that there is no need for password when accessing a website automatically using short range communication as will be explained below.

According to one embodiment, if this is the first time the authentication server (112) is run and attempts to connect to the pointer server (114), the pointer server might request the authentication server to identify itself and create an account. In other embodiments this is not required. The authentication server (112) will be supplied with the IP address of the nearest pointer server in advance.

According to one embodiment, at this stage the authentication server (112) requests the pointer server to register the mobile phone device's (110) 42 bit unique BD_ADDR number against the IP address of that authentication server (112). This request uses the low-level data protocol TCP/IP (or optionally HTTP/XML or a substitute protocol). The IP address of the authentication server could be found from the TCP/IP request header.

The process will be successful only if no one else has already registered this BD_ADDR number. If there are any pointer servers nested above, then the uniqueness of the number needs to be checked all the way to
the top (world-wide) pointer server. This step enables system integrity and stability as will be explained in more detail below.

Another option would be not to have a pointer server, but instead to transfer the Unique ID using an alternate method (e.g. SMS, MMS, email, WAP push, voice, manual input etc) for future short range communication use. One example would be cash point terminal for particular bank where once activation SMS will be sent to mobile phone device containing Unique ID, software will always know when in short range communication with bank’s cash point which bank’s IP address to contact (if needed) and which Unique ID for particular bank’s (cash point’s) BD_ADDR or BSSID or it could change own SSID (name) according to pre arranged software design including information supplied in an alternate method mention above.

Once a mobile phone device has been registered with pointer server, the authentication server must release the BD_ADDR number from it before any other authentication server or user or entity could use that number on this particular system with pointer server. Once registration is successful, the pointer server (114) will acknowledge the authentication server (112) using a data connection (113).

After each participating device, such as (110) and (116), has completed registration and logged onto the authentication server, (112) and (118) respectively, they are ready to participate in the system.

In one preferred embodiment, the user of a mobile phone device (110), as shown in figure 1a, will press one button which will launch the software installed on the mobile phone device and request the Bluetooth (RTM) built in radio module to use a radio connection (115) to request the BD_ADDR numbers of all devices in close proximity (E.g. up to 100 meters).

At this time or soon after it has collected all available BD_ADDR numbers, the mobile phone device (110) will register with an authentication server (112) that it is available to be bonded with another device or entity.

According to one embodiment, once one or many BD_ADDR addresses have been collected from nearby mobile phone devices (116) they will be sent using a GPRS/UMTS connection (111) (or substitute service) to the authentication server (112) which will send a request to the pointer server (114) asking for at least the IP address of the authentication server.
responsible for the BD_ADDR numbers of those nearby mobile phone devices (118). In this embodiment the pointer server (114) will return to the authentication server (112) the IP address of the authentication server(s) (118) responsible for the nearby device(s) (116). At this stage the link between the BD_ADDR number and its responsible authentication server IP address could be stored locally for future use, and only after it ceases to be valid would the authentication server seek clarification of the current link between the BD_ADDR number and its authentication server IP address, from the pointer server.

The process described above will be repeated for any devices, such as (110) and (116), participating in a short range communication network and available to be bonded with at a particular time, even if not mentioned in the examples below.

Upon receiving information from the pointer server (114) indicating the IP address of the authentication server (118) associated with the BR_ADDR number of the device (116), the authentication server (112) will contact the authentication server (118) at the provided IP address, with data which includes the relevant BD_ADDR number of the relevant Mobile Phone Device (116), using in this example, an XML connection (119). This contact constitutes a request from the authentication server (112) to the authentication server (118) that a user of a mobile phone device (116) associated with the authentication server (118), identified by a BR_ADDR number should link with a user of the mobile phone device (110) whose authentication server is shown figure 1 with number (112). Once this request is received by the authentication server (118), it will search its database for a data entry indicating registration of such a mobile phone device and for data indicating whether the device’s (116) user is available to be bonded with.

It is possible to register the availability of device (110) with a pointer server (114), rather than an authentication server (112) but in this instance the user data would be stored on the same centralised server and/or database used for managing the BD_ADDR number registration. This would not be the best solution for many entities, as they would not be willing to store their most sensitive data remotely using someone else’s hardware or software solution.
Also in, for example, a small company, the same physical server machine could be used as a pointer server and an authentication server. Note therefore that here, the term server means software operating on hardware. The two servers are separate, but may in some cases operate on shared hardware.

If a mobile phone device (116) is registered with the authentication server (118) and available to be bonded with, then the authentication server (118) in this example will reply to the authentication server (112) with the Unique entity ID for this user, user's name and, optionally the user's picture, registered by the user of the mobile phone device (116) at the current time.

The Authentication server (112) will store and forward the Unique Entity ID in its database and forward the name (and optionally the picture) of the user currently using the mobile phone device (116), to the mobile phone device (110) where it will be displayed on the screen of the mobile phone device (110), and be available for selection by the user of the mobile phone device (110). It would be possible to have many users displayed on the screen with or without pictures, if many BD_ADDR numbers had been collected during the search phase. This could happen for example during a busy business exhibition where there are many devices in bonding mode.

The process above could be repeated automatically for a couple of minutes until a user becomes available. Upon seeing a picture and user's name displayed on the screen of the mobile phone device (110), the user may select it and could be offered a selection of his own links including but not limited to private, business, charity and/or hobby etc. This action defines the type of link with this particular user (or more precisely defines the data to be available to the other user after the linking process has finished). The available data could be contact data, pictures, video etc, but also other data sets, for example access permissions, as will be explained below. After the selection is made on a mobile phone device (110) it will be sent to the authentication server (112) which will store this information against another user/entity ID in its database which would effectively make particular data set(s) of user of mobile phone device (110) always available to the user of the mobile phone device (116) and the authentication server (118).
After this step it will send confirmation of the link together with a Unique Entity ID for the user of the mobile phone device (110), containing Object ID pointing to all necessary information on authentication server’s (116) database and send them to the authentication server (118). At this point, the server (118) could send a request to the mobile phone device (116) and await authorisation to accept the type of link from the user of the device (110), or it could simply accept it automatically from the authentication server (112) and optionally send confirmation to the mobile phone device (116). At this step, the authentication server (118) will write in the database (Link), Unique Entity ID.

At the same time the authentication server (118) will forward the User’s name and optionally picture to the mobile phone device (116) and await selection of a user and a type of linking. Once this information is returned to the authentication server (116) it will be saved in the database and sent to the authentication server (112) with included Unique Entity ID for user of mobile phone device (116), which will be stored in the database of the authentication server (112) and optionally await authorisation from the user of the mobile phone device (110). After authorisation is received (if required) both authentication servers will synchronise data attached to received links, and make this information available to mobile phone devices linked thereto.

After this step, bonding is completed and both mobile phone devices (110), (116) are removed from the list of available bonding devices on their authentication servers (112) and (118) respectively. If a user is removed from the available list, no other arbitrary entity can request access, even of the user’s basic identification including but not limited to the user’s name or picture, which is typically available during the bonding phase.

If for example a user of a mobile phone device (116) wants his data to be discoverable by others he could select the required data to be published by the authentication server (118) to an Entity Name Server (121) using, for example an XML connection (120). The Entity Name Server (121) might need the authentication server (118) to register with it before authorising such publication. Typically during registration the authentication server (118) will be provided with its own unique ID, authentication details and/or own IP address,
which could be taken from a header of a TCP/IP request to a centralised Entity Name Server system.

The published data could have visible and/or hidden data sets for those searching. Every registered entity should provide a Unique Entity ID, which includes the IP address of their authentication server. The "push" model where contact data is published to an entity name server is preferred to the "pull" model where the entity name server must periodically search for such data on all available authentication servers. Data will be published to an entity name server by a 'push' mechanism rather than a 'pull' mechanism only if there is a change in the particular data set so the entity name server is updated by authentication server practically instantly. A push mechanism is also a preferred model for when data is changed and synchronised between authentication servers.

Users' of the system could use visible and hidden data sets on the Name Server for a users' logon and authorisation to any connected system. For example any website server, mobile phone device or other device could connect to an entity name server (121) offering only two fields. Firstly a "user name" field which could be dynamic and secondly a "public password" field for logging on. The first time a user registers for a service with a website he would be required to provide his user name, which could be dynamically resolved to a Unique ID consisting of an authentication server IP and an Unique Entity ID from data published to the entity name server (121) using, for example Asynchronous JavaScript and XML (AJAX), effectively enabling a user to type in any string of the words, and in real time reduce the number of users associated with that collection of words to only one. The collection of words could be made of visible and/or hidden published words for a particular user. Purpose of hidden words is so no other system participant can see confidential information including but not limited to another user’s postcode or try to mimic dynamic resolution of another user's data to own Unique Entity ID.

Once the collection of words has resolved to only one user (referenced by a Unique ID) of the system, for example using "Dave" as the visible word and "SW191AA" (the postcode) as hidden, the user would be required to type in the public password. The words “Dave” and “SW191AA” would resolve to a
Unique ID supplied by the server (121) using AJAX, and together with the public password provided by the user will be sent back to the authentication server (118) asking for confirmation whether this is the user of this Unique ID. After the website has a positive identification, the Unique ID could be stored in its local database for future reference, dependent on a service provided. For example if it were a web based email service, it could request from the authentication server more user data including but not limited to all names of the user’s linked contacts, to dynamically populate the service provider’s database. Only after the user starts sending email to a particular user, an email address of that user would be provided by the authentication server (118). This methodology could enable the user to seamlessly access his own data and third party services or the system itself by using a single access password for all services on the Internet or on other devices including but not limited to mobile phones.

Note public password is only needed if a device used to access website is not linked to the Unique Entity ID and/or it does not have a short range communication to authenticate user’s mobile phone device which will be explained in more details in section Web Access below.

Note that this website’s public password does not need to be the same password used by the user to access and change his own data, which is called a “private password” and should be closely guarded.

This same feature for seamless logging on, via the Entity Name Server could be used using a private password if for example a user has lost his mobile phone, had it stolen, bought a new one or borrowed one from a friend, he could log onto his authentication server and have all his contacts, linked data and other information instantly available.

This is an important step which allows a user to log on from any other mobile phone. For example if his phone is lost or stolen, that phone will be automatically logged off and all data blocked, including access to other services described in more detail below. In the future, if a user changes his own contact data using his mobile device (110) via his authentication server (112), it will notify all linked users about the data change. In this example there is no need for data to be refreshed periodically using data connection (117) to the mobile phone device (116) but only if the user of the mobile
phone device (116) tries to open an address book with specific details of the
user (110), as it would not be necessary to refresh rarely used contacts,
especially if pictures or video is used in the link of such a contact. Otherwise
this could significantly increase the data transfer through the data connection
(117), but data access speed would be increased, as there would be no need
to wait for the authentication server’s reply.

In addition any user can remove themselves from another user’s
contact list at any time should they wish.

In another embodiment similar to the above described, there is the
difference of using only two mobile devices (150, 152) represented in figure
1b without a separate authentication server or pointer servers but using a
static 128 bit IP version 6 address (which would look like
2abc:0db8:85a3:08d3:1319:8a2e:0370:7664 but for brevity this document will
use only 4 hexadecimal digits). The mobile phone devices (150, 152) will use
128 bit static IP addresses represented with shortened hexadecimal numbers
1a1a and 2b2b respectively. Devices (150, 152) can still publish their data to
the Entity Name Server (not represented in figure 1b) as they include the
authentication server functionality internally.

As in the system represented in figure 1a, users will install appropriate
software on their devices and register with the service before using the
system, after which their Bluetooth (RTM) name visible to other devices will be
Dave@1a1a and Bob@2b2b where 1a1a and 2b2b represent shortened
128bit IP addresses. The words Dave and Bob are free static text and have
no use in the system, but could be used when two users are exchanging files
directly to identify one another’s devices. Static words could be separated by
some reserved symbol like “@” or “#” which could indicate whether a device
has a dedicated Internet connection or not. This will be explained in more
detail in the next example, as represented in figure 1c. In figure 1b both
devices have a dedicated Internet connection and thus both device names
contain the symbol “@”.

As the system in figure 1b has no external authentication servers, both
devices are acting as authentication servers, where (155, 157) are optional
back-up servers synchronised using the Internet connection (154, 156),
possibly using the SyncML protocol.
A Back-up server is graphically represented with a symbol such as the one next to the number (157) in figure 1b.

After both users of the mobile phone devices (150, 152) have pressed the respective bonding buttons on their devices, their respective systems will enable Bluetooth (RTM) modules, with the names of the devices set to Dave@1a1a and Bob@2b2b respectively, and write in their local database, data indicating availability to be bonded.

After the mobile phone device (150) has read from the short range communication (151) Bob’s unique ID 2b2b, it will use a dedicated Internet connection (153) to contact Bob’s mobile phone device (152) and request further details (including but not limited to a name and picture of the user of the mobile phone device (152)). At this stage, Bob’s mobile phone device (152) will be in bonding mode and thus it will reply using its Internet connection (153), with Unique Entity ID, the name and picture of the mobile phone device user (152) which will be displayed on the screen of the mobile phone device (150) ready for Dave’s selection. After Dave has selected a picture, he will select the type of linking he wants to use with Bob i.e. which datasets are to be synchronised between them, for as long they are linked. This information will also be forwarded to Bob’s mobile phone device (152) after this mobile phone device (152) has read from the short range communication (151) Dave’s Unique ID 1a1a it will use a dedicated internet connection (153) to contact Dave’s mobile phone device and request further details including but not limited to unique entity ID, name and picture, which once received will be displayed on-screen awaiting Bob’s authorisation, or it could be automatically stored in the local database.

After the mobile phone device (152) has read from the short range communication (151) Dave’s unique ID 1a1a, it will use a dedicated Internet connection (153) to contact Dave’s mobile phone device (150) and request further details (including but not limited to a name and picture of the user of mobile phone device (150)). At this stage, Dave’s mobile phone device (150) will be in bonding mode and thus it will reply using its Internet connection (153), with Unique Entity ID, the name and picture of the mobile phone device user (150) which will be displayed on the screen of the mobile phone device (152) ready for Bob’s selection, after selection is made he will be presented
with options of types of linking available. After his selection is made this information will be stored in a local database and confirmation of the link created and will be send to Dave's device awaiting authorisation if necessary, after which both devices can disengage bonding mode and synchronise the necessary data set(s) if not synchronised already.

In some instances there will be no data sets, if an exchanged Unique ID is not linked (for example to contact data), but if instead access authorisation is provided for some service or device.

Figure 1c represents a hybrid system using an external authentication server (174) responsible for a device (170) and a mobile phone device (172) functioning with an internal authentication server responsible for its own authentication. A user of a mobile phone device (172) is requesting access from a device (170), which could be his company car. Note that the device/car (170) has neither dedicated Internet connection nor direct connection to its authentication server.

A back-up server is represented next to the number (176).

Before being able to use the system, it needs to be initialised. In this step the device (170) will become linked to an Unique Entity ID using the authentication server (174) (described above in more detail), and if using a Unique ID number without an IP address there could be a need for a pointer server.

It is possible to insert a 64 bit number identifying a user, an object or even an authentication server IP in to the 64 bit host part used for a 128 bit IPv6 (or even make it change over time). For recommendations on randomness creation consult document RFC1750 from Network Working Group available at. http://tools.ietf.org/html/rfc1750#ref-GIFFORD and for suggestions about changing IPv6 consult RFC3041 "Privacy Extensions for Stateless Address Auto configuration in IPv6" available from http://tools.ietf.org/html/rfc3041#ref-RANDOM . Noting that a random number generated could be used to identify an entity or a device or used as a link between two different entities or devices or, as in this example, it will be used to give access to car. The above references are incorporates herein by reference.
Device (170) could be pre-programmed with access numbers/codes which will be transferred/exchanged using short range communication, which unlocks the car during the manufacturing process or it could be generated by an authentication server (174) and transferred to the device (170) 'on-the-fly' during initialisation.

Additionally, the device (170) could be supplied by the manufacturer or an authentication server (174) with one or more so-called 'hash keys' used for data encryption. It might be important to encrypt data from being misappropriated from device (170) using hash functions including but not limited to SHA-1, MD5, or RIPEMD-160 with the hash key supplied, especially if authentication will be via a remote device, in this instance a mobile phone device (172) using virtual authentication by means of tunneling or encapsulation.

However encryption might not be necessary for many uses, as for example access could be granted via the 64 bit host part of an IPv6 number. The system could generate 100,000 random 64 bit numbers, which could be each be used only once to access the service and only one will work at the time i.e. not all 100,000 combinations will give access at the same time but only one, after which their validity would expire. Additionally for security purposes, the device could be pre-programmed to accept only one such number 64 bit host part IPv6 number per minute for every 64 bit network prefix with maximum of, say, 100 trials and errors and thus reduce dramatically the chances of success of a brute force attack.

However to simplify this description we will use the IP address and the Unique ID inserted in a Bluetooth (RTM) name. Specific IDs could be pre-programmed into the device (170) to unlock the car on once-only bases, with expiring times, thus eliminating the need for encryption of the data between the device (170) and the authentication server (174).

In the future (and thus it is not mentioned in this document) new data fields may be created by hardware or software manufacturers of short range communications (Bluetooth (RTM), WiFi, IrDA etc) modules for the specific purpose of passing a Unique ID between devices. One of the characteristics of the new field would be a very fast discovery of near-by device.
In addition to the initialization described above, the device (170) and the mobile phone device (172) will be linked (i.e. device (172) will have access to (170) which will be stored on authentication server (174) database), prior to full use of the system, as will be explained in more detail in the section related to the "Business/Charity link" below.

Both devices (170, 172) will be using 128 bit IPv6 and a naming convention of four hex decimal numbers abbreviation for each 64 bit part of 128 bit IPv6 address, with the addition of the host part of the IPv6 address. The mobile phone device (174) and Bluetooth (RTM) name will be set to John@1c1c.5c5c where "@" indicate the internet connection, 1c1c is the 64-bit (sub-) network prefix of IPv6 address of the mobile phone device, in this scenario, included on device the authentication server (172), and where 5c5c is the 64-bit host part of IPv6 address set to be Device ID. The device's (170) Bluetooth (RTM) name will be set to Car15#2c2c.8c8c where symbol the "#" indicates that there is no Internet nor direct data connection to an authentication server for this device, 2c2c is set to be the 64-bit (sub-) network prefix of IPv6 address of an authentication server (174) and 8c8c is the Device ID identifying the device (170).

Upon searching for nearby devices, the mobile phone device (172) will read the Bluetooth (RTM) names available, in this instance it will discover Car15#2c2c.8c8c, and at the same time the car's Bluetooth (RTM) module will read John@1c1c.5c5c and remain locked.

After this step, the user's mobile phone device (172) will parse text from the Bluetooth (RTM) name, and recognise the symbol # (meaning that specific device - in this case the car - has no internet connection available), so it will use its own connection to contact the authentication server (174), using the extracted IP number 2c2c and 8c8c Device ID from the Bluetooth (RTM) name and of the device (170), and provide its own IP address 1c1c and Device ID 5c5c (or Unique Entity ID related to this device) of the Mobile Phone Device (172).

After the authentication server (174) has positively confirmed an authorised link between the user of device (172) and (entity using) the device (170), it will issue one of the unique numbers (unlocking keys) 1f1f in this...
instance programmed in to the device during initialisation or registration, which will unlock the car once only.

Upon receiving this number, software on the mobile phone device (172) will rewrite the Bluetooth (RTM) name for the device to be John@1c1c.1f1f which, when read by device (170), will result in unlocking the car.

Additionally after this step, device (170) could change Bluetooth (RTM) name according to a pre-arranged scheme (established at initialisation) so the next time new values are past to the authentication server it will be updated accordingly with information on who or which unlocking key issued by authentication server (174) had previously successful access using that unlocking key, effectively guaranteeing only one device ID (unlocking key) at the time can unlock device (170), and for administration purpose so authentication server (174) can store data identifying which user at which period of time has accessed the device. For further security this cycles of exchanging unique Device IDs could be repeated.

Note that authentication server (174) will identify original 8c8c Device ID as one before 1f1f – which is the next unlocking key, thus although there could be 100,000 64-bit unlocking keys (codes) pre-programmed to device only one at the time in sequence will unlock device depending on previous Device ID.

Above example is with a “static” list of unlocking numbers while even better a “dynamic” solution is described below. The solution would randomly generate unlocking keys on-the-fly and very often change it through the time as it can be repeated infinitely for added security. For recommendations on randomness creation consult document RFC1750.

The solution could include an algorithm including but not limited to SHA-1, MD5, or RIPEMD-160 and a hash key (randomly generated unlocking key) as it would take less memory on the device and less data transfer during initialisation than a long list of for example 100,000 unlocking keys and it would be an infinite source of unlocking keys.

For example if there was no authentication server (174) and device (170) is trying to get access to device (172) than both devices would set Bluetooth (RTM) names as Device170#8c8c and Device172#5c5c where
8c8c initial Device ID for device (170), and 5c5c initial device ID for device (172).

Both devices (170,172) will use the same SHA-1 algorithm and device (172) has generate random Device ID (5c5c) as result of combining a noise random generated number (also used as unlocking key in this example) (1f1f) and SHA-1 algorithm. Noise random number could be easily generated from number of sources including but not limited microphone camera, hard disk etc.

Once device (170) gets number (5c5c) it will use the same SHA-1 algorithm and resolve it to (1f1f) initially generated on device (172) using noise random generation number. Upon setting this number as Bluetooth (RTM) name Device170#1f1f and once it is read by device (172) the device (170) will be granted access.

The similar method to above could be used in an example where (172) is cash point terminal with own authentication server included and linked to a Unique Entity ID on authentication server (174) using random generated number from the noise using the same hashing algorithm as device (170), and where device (170) has no dedicated access to the internet but would like to access the service provided by cash point (172) using the same hashing algorithm as authentication server (174).

Although in the previous section for describing the system the term “user” is used, the same automated process could be used to bond with a number of different services and entities. An example is in relation to billboard adverts 100 meters away using a WiFi wireless connection. After bonding, the relevant advertising company should be able to contact the user through a medium that the user has selected (E.g. from email or phone etc), and the user should have all the necessary information to contact the advertised company and to access other data including but not limited to promotional videos etc. There is no need for the billboard to have a dedicated internet connection to its own authentication server as it is used only to distribute/initiate link/bonding. While real process could be completed between user’s mobile phone device and a remote authentication server connected to the internet and related to Unique ID distributed by the billboard.

Examples of uses of the system are numerous
The system mentioned in figure 1 has a limitation of only one pointer server storing and serving all system users, which would make it in practical terms at worldwide level very difficult to run, even with sophisticated and distributed load balancing. Additionally some organisations including but not limited to governments or large businesses will be unwilling to share such internal information as BD_ADDR/BSSID numbers for bonding between two devices/users which could (although very remotely) identify two people/personnel bonding together. The proposed solution to such problems is shown in figure 2 which adds a multi-layer system providing a top-level server(s) for the whole world, with second level servers associated with countries, regions or continents, and then lower level servers for organisation including but not limited to governments, firms and service providers (and so on as needed). In such instances the bonding process would be achieved locally and only if the BD_ADDR number is not registered locally, the system would progress a search or registration one level up. This process continues until the server with knowledge of the BD_ADDR number is found, or the top level server replies that no such BD_ADDR number is registered.

Additionally to relieve strain on the pointer servers and to increase the speed of authentication, especially for access, it would be essential (where access is granted) to store the data locally on device level and periodically update a database of all authorised BD_ADDR numbers of authorised users or devices. Ideally this should be implemented using an electronic memory including but not limited to high speed RAM, rather than using hard disks with slower access times. For example in public transport each bus connected to the system should hold locally a database of all users who can use this bus transport system, rather than waiting for the authentication server’s reply.

Figure 2 shows a simple representation of such a system where a smaller system (200) similar to that described in Figure 1 is contained inside of a larger system, maintaining integrity of the system but preserving all data transfer between its users locally and reducing the strain on the larger system. The smaller system (200) is described above, but if a BD_ADDR number of a mobile phone device (207) is found by a device (116) and not located using a pointer server (114), then using an XML connection (202), the pointer server (114) or the authentication server (118) will seek identification from the pointer
server (201) and if found, the server (201) would reply with at least the IP address of the authentication server (204).

Upon receiving this information, the authentication server (118) will contact the server (204) asking if the relevant user related to BD_ADDR is available for bonding. If so, it will receive the user name and/or picture or other data, which will be made available to the user of the mobile phone device (116). Once the user of the mobile phone device (116) selects on which level bonding should be made, (including but not limited to private, business, hobby etc) this information will be passed to a server (118) which will write selection to the database and send data to the authentication server (204) making it available for the user of mobile phone device (207).

The same process as above would be repeated by another user to enable a user of the device (116) to access the data of the user of the mobile phone device (207). After successful exchange of Unique IDs users/(devices) and/or their authentication servers, will synchronise data and update mobile phone devices. Note that this process was described in more detail previously but for the sake of simplicity, this example was kept short.

The system mentioned above has no limit when adding new pointer servers using the methodology described for pointer servers (201, 114). Thus there could be a multitude of levels including but not limited to world, continent, region, government, company, city, network provider, and all coexisting in the same distributed environment, thus preserving the data of users.

The above described system is a simple example of bonding between only few people exchanging contact details directly, but it is possible to have the same principle of bonding between different people and entities and machines for many different uses as will be described below. Note that for sake of simplicity, not all details will be repeated in every example but generally only the ones which are unique for each particular case, although the principles of user/entity registration, de-registration and searching etc will remain the same.
Specific Applications

Business/Charity link (Exchange link)

This section will explain how an employee could get additional own employment contact data (including but not limited to a business name, telephone number, address etc) from a new employer. This could be distributed to other business people during the course of normal day to day business. Additionally a company which has issued an employee business contact data could decide, and issue a policy, to keep some or all contact data made during employment, if for example the employee was employed as a sales representative. Additionally an employee could simultaneously get access to the company’s office, desktop and/or car etc.

In this example the system is without a pointer server thus there is exchange of at least the IP address of the authentication server responsible for the device using short range communication preferably included in Unique ID.

Before using the system, all three users will register for the service to their respective authentication servers, thus creating databases populated with information represented therein. Examples include user John in Figure 5, rows next to the numbers (501) for Private and Public passwords and a bonding tag, for a mobile phone device (502), car (503), home (504), name (510), phone (511), email (512) and then defining a Private Link (555) by grouping previously entered information. Creation of the rows next to the numbers 556, 561 and 562 will be explained latter in the document. Similar methodology will be repeated for users of the databases represented in figures 6 and 7.

Initially an employee will register at his company’s security office, using his mobile phone device (318) as in Figure 3. At the company’s security office there is provided a desktop computer (323) with a Bluetooth module for short range radio communication (322) used for exchange of a Unique ID between an employee’s mobile phone device (318) and the desktop computer (323). Additionally the mobile phone device (318) and the desktop computer (323) are connected using the Internet to authentication servers (316, 314)
respectively which are further connected together via an internet connection (315).

The unique ID in this example contains a 32 bit IP address of the authentication server responsible for the device, a Unique User ID and an Object ID.

For example if the IP address of the authentication server is 123.123.123.123, the User ID is 55, the mobile phone device Object ID is 01 and the user’s name is John, his Bluetooth (RTM) name could be set to John@1231231231235501. Note that port number or other information could be added using the same methodology. The name before the @ symbol could be anything (as chosen by the user) and is not used by the system. The number after the @ symbol becomes a Unique ID which could be further compressed by encoding a hexadecimal number of an IP address and possibly of a User ID and an Object ID for this particular server and/or database or even using alphanumerical coding from some character set for full use of all bits available in every byte.

Pressing a specific button for linking (or choosing an option) on the mobile phone device (318) will activate the Bluetooth (RTM) module, setting the device name to John@1231231231235501 and be visible to other nearby devices. After this step it will start a search for other nearby Bluetooth (RTM) names. After the mobile phone device (318) has collected Bluetooth (RTM) name(s) in this example BigBusiness@1001001001008005 (where 100.100.100.100 represents the IP address of the authentication server (314), the number 80 identifies the entity on the authentication server (314) and 05 is an Object ID for Security Desktop 8 (322), it will then request from the authentication server (314) the name and possibly a picture of the entity behind this unique ID.

If the desktop (323) has indicated to the authentication server (314) that it is available to be bonded by inserting a tag <BOND> in the row next to the number <706> in figure 7, then the company name, desktop name (in the case that there are more than one available in the same office) and possibly company logo will be returned to the authentication server (316) and forwarded to the mobile phone device (318) or it could be directly forwarded to
the mobile phone device (318), upon which successful completion, it will be displayed on the screen of the mobile phone device (318) ready for selection.

The same steps from the above paragraph will be repeated for the desktop (323) upon which successful completion, the name and picture of the mobile phone device's employee/owner (318) will be displayed on the screen of the desktop (323) ready for selection.

After the employee of the mobile phone device (318) has selected the name and possibly company logo on the screen of the device resolving to Unique ID 1001001001008005, a new screen will open with a selection of predefined links. In this scenario the employee will select "Private Link" which will result in a Unique ID 10010010010080?? being stored in a row next to the number (555) where "??" represents any number. Thus any object from this particular entity could access information inside next to the number (555) object, resolving in this particular example to the Name, Private Mobile Phone Number and Private Email Address of this employee. If there is need to restrict access to only a particular object from the requesting party this could be implemented by specifying the Object ID numbers in the authorised field of the database.

This unique ID 1231231231235521 will be sent to the authentication server (314) and then forwarded to the desktop (323) awaiting the security officer’s approval. Once approval is granted, a new row in the database on the authentication server (314) will be created next to the number (777) with a Unique ID 1001001001008055 and a Unique ID 1001001001008021 stored in the “Data” column of the database, thus making a link between those two objects which could have their contact data transferred or synchronised as necessary. At this stage the employee’s authentication server and mobile phone device could be notified of a successful link creation.

After this step a security officer will select an employee’s name and possibly picture resolving to Unique ID 1231231231235501, and additionally the type of linking - in this case “Executive Business Link”, which will result in the creation of a new row next to the number (781), creating access rights for the desktop 12 placed in the office, the Company Car 15, access to the company’s building and all doors necessary for the employee to do day-to-day business, and permissions for access to a company parking place
by populating the database column “Linked/authorised” with the relevant access Unique IDs in the “Data” column and the Unique ID from the employee’s mobile phone device (318) Bluetooth (RTM) name 1231231231235501 which will be stored in the column “linked/authorised”.

Additionally a Unique ID 1001001001008021 linking to business contact details will be created and sent to the employee’s authentication server (316). Note, because the company’s policy in this scenario is to hold any contact data the employee has gained during normal business duties in the name of the company, and to withhold such data. As a result any business data or contacts/links gained are stored on the company’s authentication server (314). In practice the tag <OWN> could be inserted in the request body so that the authentication server (316) can place the Unique ID together with other links used to link with other entities including but not limited to “Private Link”, “Business Link”, “Hobby Link” etc. Once approved by a user of the device (318) this will be stored in the row next to the number (556) and authentication server (314) and desktop (323) are accordingly notified of success.

At this step or afterwards, data can be transferred and/or synchronised as necessary between the employee and company, and both the mobile phone device (318) and the desktop (323) remove themselves as devices available to be bonded, on their respective authentication servers.

Another feature of this example will be that once the employee has been granted credentials or access rights from the company it could automatically enable him to access company’s desktop computers, unlock automatic doors of the office building or office or even use a company car with a module which is connected to the internet (for an example without the internet connection please see figure 1c description) in the same fashion as any mobile phone device is connected to the authentication server and a Bluetooth (RTM) module name used for authentication in conjunction with an electronic central locking with a Bluetooth (RTM) module and connection to an authentication server. An example of business access for the registered user above will be briefly described below.

In this example the same Figure 3 will be used, but with a different purpose, when a registered employee, using a mobile phone device (318)
comes in a close proximity of a company's automatic doors (323), which has a Bluetooth (RTM) module for short-range communication (322) and an intranet connection (324) to the authentication server (314). The mobile phone device (318) will have its Bluetooth (RTM) module constantly active, and its name visible, as described above, and the automatic door (323) in this example will constantly send a request for discovery of any nearby devices in the range of 10 meters. Upon receiving any new Bluetooth (RTM) names (Unique IDs) it will search a local database, and if not found, it will request from authentication server (314) authorisation, and if granted it will unlock the door.

Authorised Unique IDs should be stored locally to the electronic door so that the access time is lower, however it should be synchronised periodically so that the database is refreshed if an employee has lost a mobile phone or if employment has been terminated or on non payment of fees (in an example of public transport). Additionally, periodically the authentication server (314) could request periodically from another authentication server (316) that the data is still correct and this Unique ID is still registered against a particular user or entity. This method is called "pull". Another preferred method would be that if there is any change, for example the user's mobile phone is stolen, in which case the authentication server (316) would notify immediately authentication server (314) to remove Unique ID permissions immediately from its systems.

The second part of this scenario will be described by way of an example of exchange of links between two user's (John and Joe) using Figure 3 as a schematic of the embodiment and figures 5, 6 and 7 as simplified versions of their databases.

Both users of mobile phone devices (318) and (310) have their Bluetooth (RTM) names set to John@1231231231235501 and Joe@0990990990994001 respectively are using the same naming convention as before and in the case of John the same database represented in figure 5. In this example Joe is pre-registered with authentication server (312).

After both users have clicked the bonding button (or choose an option) on respective mobile phone devices their respective authentication servers will register this information by adding the tags <BOND> in the fields next to the numbers (501) and (601) respectively.
Separately both devices will collect nearby Bluetooth (RTM) names and send them to their respective authentication servers which will, in turn, contact each other with expectation of receiving a name and possibly a picture of the respective user.

Once the names and possibly pictures are received, they will be displayed on devices, upon which users will make a selection. Immediately after selection of the particular user on the screen another screen will open asking for the type of linking, where the user of the mobile phone device (318) will select the “Private Link” and “Business Link” options. Accordingly this information will be sent to the authentication server (316), which will send the Unique ID 1231231231235521 and the unique ID 1001001001008021 to the authentication server (312) using the internet connection (325).

Note that the unique ID 1001001001008021 links directly to the company’s authentication server (314) as a result of company’s policy and the authentication server (316) will send a request to the business authentication server (314) to enable the user of the mobile phone device (310) with unique ID 09909909909940?? to see his business contact data, which results in the Unique ID being added in “Linked/Authorised” column in the row next to the number (720). After employment of user of mobile phone device (318) has terminated, the company will continue to have access to critical data including but not limited to contacts that the employee has made during the employment such as contact details of Joe in this example.

Upon receiving these two Unique IDs, the authentication server (312) will send it to the mobile phone device (310) for authorisation and if granted, it will create two rows next to the numbers (661) and (662) and populate them with respective unique IDs received.

After the user of the mobile phone device (310) has selected a name, a picture and a type of linking (in this example Private Linking) Unique ID 0990990990994021 will be sent to the authentication server (312) , then forwarded to the authentication server (316) and then forwarded again to the mobile phone device (318) for authorisation. This, once granted, will result in authentication server (316) creating a row next to the number (562) and storing the Unique ID 0990990990994021 in the “Data” column.
After all the steps are completed, the devices will be removed from bonding mode, and all necessary data transferred if it is not already.

Note that the above principle of linking is designed with separate database rows for incoming and outgoing connection, thus making it possible for one of the users to delete an incoming contact link of another user, but still stay visible to another user through outgoing link. Whereas if there was only one row for incoming and outgoing connections, then deleting from each end will terminate both links.

**Currency transfer between two users who have never met, or are already connected**

This section explains how currency or other information could be securely exchanged between two users. Note that as it will be apparent from this example, currency could be valid between only two people or world-wide only depending on user’s choice and trust.

After registering the first user Unique ID using the mobile phone device (318) in a Bank’s branch against his bank account using methodology previously explained, where his bank account becomes an object related to bank and this particular user, a user may exchange currency or information. In this scenario the bank’s desktop computer (323) has a Bluetooth (RTM) module for short range communication (322) and an Intranet connection (324) to the banks authentication server (314).

After registration, a user can hand over cash, which will be registered against his Unique ID in the bank using an authentication server (314) and a database attached to it. The same process will be made by a second user of a device (310). It is important to note that the process of registration in the bank, as described above, will add a Unique ID pointing to the bank’s authentication server, to a mobile phone device (318) and (310) and their respective authentication servers (316) and (312) similar to a business link or private link but which will never exchange any data (including but not limited to a bank’s account name or details), between individual users when selected, but instead will point the mobile phone device(s) to a bank’s authentication server for verification and authorisation. In this example for sake of simplicity a single bank is used but in practice it could be a network of inter-linked banks.
After two users of mobile devices (318, 310) who have never met before, come in close proximity, they will be able to press a button on mobile phone device which will make their respective devices available for bonding - in this case to enable currency transfer, which will update their status on authentication servers (316) and (314) respectively. As a result they will be visible to mobile devices in close proximity using the previously explained methodology for bonding between two users. However once the users come to the option of selecting links including but not limited to Private, Business, Charity, Hobby etc, there will in this case be a Bank’s link as well.

If exactly the same bank and/or network of the banks is selected by both users (optionally for example a newspapers retailer could have his mobile phone device always in receiving mode with a specified amount for a very fast transfer). Another screen on a display of a mobile phone device (318) will open requesting to enter an amount to be transferred and optionally a password for larger amounts. Once this step is completed, data will be sent to the authentication server (316) which will forward a unique (entity) ID of the recipient, unique (entity) ID of sender a password (if required) and an amount to be transferred to the bank’s authentication server (314). Thus requesting to transfer the amount of money as specified by the user of the mobile phone device from the bank account of the user using mobile phone device (318), to the bank account of the user using mobile phone device (310) which could be found from their respective Unique IDs collected during the bonding procedure and resolved to unique entity ID using authentication servers as described previously.

If there are sufficient funds, the money will be transferred to the bank account of the user (310). In practical terms this means writing to the bank’s database about the transfer, after which, both users could be sent a notification. Note that it is possible to have both mobile phone devices (318, 310) pointing and dealing directly with the bank’s authentication server (314), for speed of transfer using a data connection (318) but this is a less secure option.

The above example is for users who have never met before, but another principle would be to select anyone from a phone book and then choose the option to transfer money. At this stage, a mobile device (318) will
send data including an amount of money to be transferred, a recipient's unique ID (from the phone's contact links) as well as any own authentication data necessary like unique entity ID for the transaction (including a password if necessary), for the Bank's authentication server (314) via user's own authentication server (316).

In all examples above additionally every request should be double-confirmed (backwards to originator) with respective user's authentication server, and the mobile phone device requesting the transfer, before allowing the transfer so that unauthenticated requests with false Unique (Entity) IDs may be avoided. This is due to the fact that short range communication and node device could be manipulated but not all of the internet network.

**Identification between two users by a third party (Police ID)**

On the same principle as above, where two mobile phone devices are pointing to a third party authentication server, it is possible to verify the identity of a mobile phone device user. For example, one person could be a resident of a particular country using a mobile phone device (318), and a second person could be a police officer using a mobile phone device (310), where after both resident and police officer have pressed a bonding button on their respective devices, and selected a 'Police Identification link'. This would result in both authentication servers (316, 312) changing the status of their users, in this case the resident and police officer, to be available for this type of bonding, and consequently both pointing to a police authentication server (314).

The police officer's and the resident's images and names would be revealed during the initial bonding stages where both participants could select each other's identities to see more details. As a result, the resident would see identification of the police officer, and the police officer would see identification details and other information of the resident supplied by the authentication server (314). It would be impossible to forge this, as both devices would be pointing to the same trusted server, in this case the police authentication server (314) preset in advance with the resident and the police officer using methodology previously explained.
In another scenario, still using figure 3, a similar principle could be used for passport control, where a desktop computer (323) would automatically collect Bluetooth (RTM) names of mobile phone devices (310, 318) of a border control officer, and a citizen crossing a border respectively. On prompting by a border control officer, a desktop computer (323) would unlock and display relevant information for the border control officer, and available to a government authentication server (314) for a particular traveller, including additional authentication information (which may be biometrics including but not limited to iris scans or finger prints) if necessary.

Web Access

In yet another example it will be described how the same system could be used to provide automatic secure access to a website (for example of some bank with additional security), potentially eliminating any need for logging a username and/or password (unless a very large amount of money is being transferred, where an additional password or biometrics might be requested for an additional security).

Once a user of a mobile phone device (318) has come into close proximity of a desktop computer (323) in an Internet coffee shop with a Bluetooth (RTM) module enabling short range communication (322) it will be possible for the desktop computer to collect the Unique ID of mobile phone device (318).

At the same time, a user will press his respective bonding button, and select a desktop he wants to connect to, the Unique ID of which will be sent to his authentication server (316). From this point if the request comes from the Desktop's (323) or authentication server's (314) particular Unique ID to the authentication server (316), it will confirm with the mobile phone device (318) whether it is still in close proximity of the desktop computer (323). If so, it will return positive confirmation. If not, it will stop providing any further confirmations to the authentication server (314) and return negative confirmation.

Once a user connects to a secure bank's website running for example 128 bit SSL (or other) encryption where a local script is executed on the desktop computer (323) (for example Active X or some other technology), it
would request Bluetooth (RTM) names containing Unique IDs of devices in close proximity, in this instance a mobile phone device (318). Upon receiving the Bluetooth (RTM) name it will be sent using Internet connection (324) to a bank's web server and authentication server (314). Upon parsing the IP address of the authentication server (316) from the Bluetooth (RTM) name, the authentication server (314) will use the data connection (315) to request from the authentication server (316) to positively identify that mobile phone device (318) as being in close proximity of the desktop (323) and having this Unique ID and return unique entity ID if different from unique ID.

If the user is in close proximity of the desktop computer (323) and is positively identified with a Unique Entity ID the same as a previously registered user for this service, the secure website would automatically unlock. Once a user has left the desktop computer, and his mobile phone device is not in close proximity any more confirmed by mobile phone device or desktop or both, the bank's secure website would automatically restrict access to some or all of its functionality.

Election Functionality

In yet another example it is possible to have an election within a company or on different government levels including but not limited to local, country-wide or internationally.

Once each citizen has registered their details against a unique entity ID and an authentication server IP address has been linked to a government authentication server, (which could be used additionally by police or by passport control) it is possible for the authentication server (314) which is part of a government to send or push an election question to registered citizens' mobile phone devices (318, 310). Upon receiving the election options or questions they could reply directly to this authentication server (314) (or more securely via authentication servers (316, 312) respectively).

If the authentication server (314) has positively identified a user's current Unique Entity ID as previously registered for a citizen using the particular Unique Entity ID registered with the government authentication server (314), their vote or reply would be accepted.
Peergroup TV – mp3/mp4 to home/business, music/video handover

This section will explain how any home or business desktop computer or screen including but not limited to a TV or display may be linked with any remote storage device, (for example for storing pictures, music or video or any other content) and connected via the Internet or an intranet.

Firstly, before use, a remote video storage device needs to be uploaded with video content, and linked to a user’s Unique Entity ID stored on an authentication server database (316). Typically a link on the authentication server (316) side would include an IP address of a remote storage server database (312) if it were not the same as the authentication server’s IP address, and any additional logging information as necessary. Additionally this could be added to a list of a user’s own links (including but not limited to Private, Business, Hobby etc). The difference is that by selecting, for example a Video link, no contact data need to be exchanged with a receiving party as with a Business link, but there could be a time limit to restrict access with other bonded devices using this link.

Once a user’s remote Video storage is defined and a Video link added to a user’s list of own link’s on a mobile phone device (318), this user would be able to press one button while at his colleague’s home and see his colleague’s networked TV (323) in the list of devices available to be bonded (as this TV is always in bonding mode waiting for the specific type of bonding as explained above). Please note that the TV (323) is connected to the Internet and the authentication server (314), where the TV is linked to a unique entity ID of the owner of the premises (in this case a colleague). The TV has a Bluetooth (RTM) module enabling wireless exchange of Bluetooth (RTM) names using short range communication (322).

After selecting a colleague’s TV (323) from the list of available devices on mobile phone device (318), a user will select the relevant Video link on the mobile phone device screen, which will result in a request to the Authentication server (316) being sent using a data connection (317). The request will include a Bluetooth (RTM) name of the TV device (323) as well as a Video sharing Link request. Upon receiving this information, the authentication server (316) will request from the authentication server (314) to
issue a temporary access for the TV (323) for the user of the mobile phone device (318).

At the same time the authentication server (314) will confirm with the TV (323) that the mobile phone device (318) is in close proximity using the Bluetooth (RTM) name of device (318) and additionally confirm with TV owner if it is OK to allow user of mobile phone device (318) access.

If positive confirmation is received, the authentication server (314) could connect directly to the Video database (312) or via the authentication server (316) (which is overseeing the authentication process to restrict content).

Once the authentication process is finished the authentication server (316) can pass Video options and Video stream handling directly to the TV (323). At this stage all options including but not limited to choice of video to be played could be commanded from the TV remote control or from the user's mobile phone device (318) or TV owner's mobile phone device. Periodically the authentication server (316) will check if the device (318) is still in close proximity of the TV (323) and, if it is not, authentication will be terminated and all access to Video content interrupted.

A similar principle could be used on a networked mp3/mpg4 player with an Internet connection and Bluetooth (RTM) or a home/car/business music/video system with an Internet connection and Bluetooth (RTM) module, where once a listener of an mp3/mpg4 player walks between home, car and office there would be an automatic switch over of music or video played on the home, car or office music or video system from a remote music or video database, using all necessary authentication servers as described previously.

**Advertising**

In yet another embodiment it is possible to connect a user of a mobile phone device (318) to a company advertising some product or service using (for example a large advertising billboard) with a WiFi module (323) with a range of up to 100 meters, and an Internet connection (324) to a remote authentication server (314).

In this particular instance it is also possible to do without a direct Internet (or other) connection (324) to the authentication server (314) and the
billboard (314), as the billboard does not need to store locally any information about the user interested in the particular advert. The only purpose is to distribute the Unique ID for this billboard, which at a particular time is related to a particular advert and unique entity ID of advertiser on a remote database stored on an authentication server (314).

After seeing an interesting advert in the distance, a user will be able to press the appropriate button on his device to initiate a bonding process and select the name of the advert, or even see a picture or video on their mobile phone device (314) identifying the advert. After this step the user will be able to select a link, (for example Private) which will cause his mobile phone device's WiFi module to request unique addresses (BSSID numbers or SSID names) of nearby devices, which once collected will be transmitted, as well as user's selection of Private link to the authentication server (316) using the a data connection (317).

Upon receiving this information the authentication server (316) will request from the authentication server (314) an exchange of contact details for a Private link. After this is complete the advertised business will have the contact details of the interested user including but not limited to his mobile phone number for SMS, MMS or Video communication attached to his Private Link and the user will have data added to his links including but not limited to the company's website URL, telephone number and/or promotional video for more information. The user can at any time change his preferences on how he wants to be contacted by the particular company, (for example by email instead of by SMS), or they can withdraw their consent to be contacted at any time.

Logical connections between the system user and devices

Figure 4 represents an example of possible logical and physical connections between a user and other devices through a network of authentication servers.

Authentication server (400) holds the data about the user and his directly related devices (401, 402, 403 and 404) which are directly linked to user's Unique Entity ID and additionally linking to other authentication servers (405, 408 and 410) and their devices (406, 407, 409 and 411).
In one embodiment the device (401) could represent a user’s desktop computer connected to the Internet with Bluetooth (RTM) and/or WiFi module, a mobile phone device (402) connected to the Internet and with Bluetooth (RTM) and/or WiFi module, as well as further examples (private house access (403) with CCTV, Security alarm, centralised heating and air-condition, electronic door locks etc), each connected to the Internet with Bluetooth (RTM) modules placed at particular places. Another examples is private car access (404) with electronic locks connected to the internet using for example standard mobile phone technology to connect to authentication server (400) via the internet and using Bluetooth (RTM) module for exchange of Unique ID, or without an internet connection using methodology used when describing figure 1c.

A business authentication server (405) stores not just information about other business contacts and relationships who its employees met through day to day business operation but also access to business desktop computers at the office (406) connected to the Internet or intranet and with a Bluetooth (RTM) or WiFi module. This could enable a user to gain access to the building and a restricted area or even a company’s car using electronic locks (407) connected to the Internet or intranet and with a Bluetooth (RTM) or WiFi module.

A government authentication server (408) is connected to a desktop computer (409), which automatically displays the information about a citizen at a passport control point connected to the Internet and with a Bluetooth (RTM) module. The same technology and methodology could be used to get access to public transport.

Another example is a bank’s authentication server (410) with a cash point terminal (411) connected to the Internet and with a Bluetooth (RTM) module. A user might be requested to enter a PIN (personal identification number) before money is ejected from the machine or for smaller transactions (like purchasing a newspaper from retailer) with a connected mobile phone device with a Bluetooth (RTM) module this might not be necessary because the risk is lower and speed of transaction and simplicity is at a premium.

Additionally to the above methodology explained throughout the document it should be noted that instead of using a BD_ADDR number or
BSSID numbers for a Unique ID there could be use of any other identification code using a wireless link. Such a number could be static i.e. never changing, or dynamic i.e. changing through time. It could be some hardware number stored on the device, or generated by the system on the server or client side. It could be identifying a device or machine, a user, another entity (including but not limited to business, government, organisation etc) or some other entity or object. The full range of combinations are not listed exhaustively herein for brevity and for simplicity of this document. Most importantly the use of different user's/entity's Unique Entity IDs or device/machine's Unique IDs than those describes does not depart from the spirit and scope of this invention.

The database representations in figures 4, 5 and 6 are an exemplary embodiment, where the Short Description column is an option and the Unique ID of the user are not necessarily the same as the Primary Key in the database.

All requests from a particular Unique (Entity) ID, even if not mentioned above for sake of brevity, should be positively confirmed with the requestor's authentication server or even with the device making the request. This is to guard against what is known as "spoofing" of a Unique (Entity) ID with the purpose of gaining unauthorised access, where a Unique (Entity) ID request does not come from the expected owner of the Unique (Entity) ID.

Any data linked, related or attached to a particular User (Entity) ID can be synchronised or transferred during the process of linking and indeed at any later stage, even if not mentioned above

Further embodiments are set out in the claims.

The apparatus and methods disclosed and claimed herein can be made and executed without undue experimentation in light of the present disclosure. While the apparatus and methods of this invention have been described in terms of preferred embodiments, it will be apparent to those skilled in the art that variations may be applied to the apparatus, methods and in the steps or in the sequence of steps of the method described herein without departing from the concept, spirit and scope of the invention. All substitutes and modifications apparent to those skilled in the art are deemed to be within the spirit, scope and concept of the invention as defined by the appended claims.
Terminology Definitions

A Database is a collection of structured data that is stored on a device so that a program or a system can consult it to answer queries. It is not necessarily stored on a harddisk, in some instances data is stored and deleted on a Flash memory module which retains data after power is disconnected, or any other volatile or non-volatile memory.

An Internet connection is usually achieved using the Internet protocol suite and is the set of communications protocols that implements the protocol stack on which the Internet and many commercial networks run. Also referred to as a long range communication through out this document.

In this document it is referred to, as a general practical term for connecting two separate devices.

An intranet is usually used for internal communication over a Local Area Network (LAN).

Bluetooth (RTM) is an industrial specification for wireless personal area networks (PANs). Bluetooth (RTM) provides a way to connect and exchange information between devices including but not limited to mobile phones, laptops, PCs, printers, digital cameras, and video game consoles over a secure, globally unlicensed short-range radio frequency. The Bluetooth (RTM) specifications are developed and licensed by the Bluetooth (RTM) Special Interest Group. Depending on class 3, 2 or 1 of the module it has range of 1, 10 or 100 meters respectively. More information on functioning of Bluetooth (RTM) can be found from The Bluetooth (RTM) Special Interest Group.

Wi-Fi short range communication is intended to cover all IEEE 802.11 standards and substitutes therefore.

The Infrared Data Association (IrDA) defines physical specifications communications protocol standards for the short range exchange of data over infrared light, for uses such as personal area networks (PANs).

Short range communication consists of Bluetooth (RTM), WiFi and Infrared standards or substitutes therefore.

XML or the Extensible Markup Language is a general-purpose markup language. Its primary purpose is to facilitate the sharing of data across
different information systems, particularly via the Internet. XML is recommended by the World Wide Web Consortium. It is a fee-free open standard. The W3C recommendation specifies both the lexical grammar, and the requirements for parsing.

AJAX shorthand for "Asynchronous JavaScript and XML," is a web development technique for creating interactive web applications. The intent is to make web pages feel more responsive by exchanging small amounts of data with the server behind the scenes, so that the entire web page does not have to be reloaded each time the user requests a change. This is intended to increase the web page's interactivity, speed, and usability.

SyncML (Synchronization Markup Language) is the former name (currently referred to as: Open Mobile Alliance Data Synchronization and Device Management) for a platform-independent information synchronization open standard. SyncML technology could be used when synchronizing data between different devices including but not limited to mobile phones, desktops, servers, terminals etc. through out this document.

OBEX (abbreviation of OBJect EXchange, also termed IrOBEX) is a communications protocol that facilitates the exchange of binary objects between devices. It is maintained by the Infrared Data Association but has also been adopted by the Bluetooth Special Interest Group and the SyncML wing of the Open Mobile Alliance (OMA).
CLAIMS

1. A system for facilitating transfer of a unique identification code between devices with the purpose of linking an entity and another entity, the system of at least two separate devices comprising:

   A first device, a wireless terminal with at least one long range communication connection and at least one short range communication connection,

   A second device, a wireless terminal with at least one short range communication connection,

   At least two separate databases, each containing at least one unique identification code.

2. The system of Claim 1 where a data stored on the database is related to the unique identification code.

3. The system of Claim 2 where the data is transmitted between the authorised and related databases.

4. The system of Claim 1 further comprising:
A first server with a database connecting and synchronising data to the first device and other servers and databases in the system, provided with authentication data relating to the first device, the first server and database adapted to relate an entity unique identification code and another entity unique identification code.

5. The system of Claim 1 further comprising:
A second server with a database connecting and synchronising data to the second device and other servers and databases in the system, provided with authentication data relating to the second device, the second server and
database adapted to relate an entity unique identification code and another entity unique identification code.

6. The system of Claim 4 and Claim 5 where a data stored on the database is related to the unique identification code

7. The system of Claim 6 where the data is transmitted between the authorised databases

8. The system of Claim 4 and Claim 5 further comprising:
A server with a database for accepting, storing and relating an IP address of the first server and the unique identifying code stored on the first server.

9. The system of Claim 8 further comprising:
The server with the database for accepting request from the second server, a unique identification code related to the first server, and responding to the second server with the IP address of the first server.

10. The system of Claim 4 and 5 further comprising:
A server adapted to receive and store information from the first server and/or the second server.

11. The system of Claim 10 further comprising:
The server adapted to respond to an enquiring device with the stored information resolved to at least a single unique identification code.

12. The system of Claim 1 further comprising:
A server adapted to periodically back-up the first device.

13. The system of Claim 1 further comprising:
A server adapted to back-up the first device upon device's request.
14. The system of Claim 1 further comprising:
The first entity related to the device has access to a services offered by a second entity related to the device.

15. The system of Claim 1 further comprising:
The first entity related to the device has access to authorised personal data associated of a second entity

16. The system of Claim 1 further comprising:
The first entity related to the device has control of a second device.

17. A networking system for facilitating the establishment of a relationship between a user and a remote device or the user of a remote device,
- the networking system comprising:

   A first system having:
   - A first device, being a wireless mobile Internet connected device being:
     identifiable by a unique identification code,
     controlled by a user,
     adapted to search for proximal wireless devices in response to a user’s input via user interface means,
     adapted to graphically represent identified devices, and
     adapted to accept a selection of one thereof by the user,
   - A first server related to the first device, provided with authentication data relating to the first device,

   A second system having:
   - A second device being a wireless device identifiable by a unique identification code, and
   - A second server related to the second device, provided with authentication data relating to the second device,

   Means for providing an IP address of the second server to the first system,
- the networking system specifically adapted such that;

On user selection to the first device, of a graphical representation of the second device, the first system is specifically adapted to:

- determine the IP address of the second server,
- send a data request to the second server, encoding a request for at least one of the first device and its user, to be henceforth authorised to access the second system,

On receipt of such a data request, the second server is adapted to authenticate the user of the first device by at least one of:

- Requesting verification from a user of the second device, via user interface means of the second device, where access to personal data relating to the user of the second device is requested, or
- Making a data request to the first server, encoding a request for verification that a user of the first device is seeking such authorisation, where access to a service offered by the second system is requested,

On verification, the user of the first device may use the first device to either:

- access personal data associated with user of the second device,
- control the second device, or
- access a service offered by the second server,

In the case that the user of the first device instead chooses to use an alternate device, the second server is specifically adapted to:

- accept remote logon by the user of the first device, with personal user data via the first device, and
- update at least one of the second server and the alternate first device with authentication data, such that the user may continue to access such personal data, service or control of the second system, via the chosen alternate first device.

18. A wireless mobile internet connected device specifically adapted for interaction within the networking system of claim 17, and specifically adapted to be able to take the role in said networking system of said first device.
19. A wireless device identifiable by a unique identification code specifically adapted for interaction within the networking system of claim 17, and specifically adapted to be able to take the role in said networking system of said second device.

20 A server for authenticating a wireless mobile internet connected device, specifically adapted for interaction within the networking system of claim 17, and specifically adapted to be able to take the role in said networking system of said first server.

21 A server for authenticating a wireless device, specifically adapted for interaction within the networking system of claim 17, and specifically adapted to be able to take the role in said networking system of said second server.

22. A server for providing an IP address of the second server to the first system on receipt of a data request encoding the unique identifying code of the second device.

23. A computer program specifically adapted for the device of claim 18 or of claim 19.

24. A computer program specifically adapted for the server of claim 20, of claim 5 or of claim 7.

25. A computer system, device, server or computer program as hereinbefore described with reference to figures 1a to 7.

26. A method for modifying an IP address of a client device so it contains a unique identification code of an entity,

The method comprising:
Setting of an network 64-bit (sub-) network prefix of an IP address to network part of an IP address locating the client device on the internet by using an IP address,

Setting of a network 64-bit host part of an IP address to network part of an IP address so it contains a unique identification code of an entity.

27. A method for modifying an IP address of a client device so it contains an IP address of a client's authentication server,

   The method comprising:

   Setting of an network 64-bit (sub-) network prefix of an IP address to network part of an IP address locating the client device on the internet by using an IP address,

   Setting of a network 64-bit host part of an IP address to network part of an IP address locating the authentication server on the internet by using an IP address
Application No: GB0710530.7  
Examiner: Gareth Griffiths  
Claims searched: 1  
Date of search: 3 September 2007

**Patents Act 1977: Search Report under Section 17**

**Documents considered to be relevant:**

<table>
<thead>
<tr>
<th>Category</th>
<th>Relevant to claims</th>
<th>Identity of document and passage or figure of particular relevance</th>
</tr>
</thead>
</table>
| X        | 1-7, 12-16        | GB2417858 A  
(ANWAR SHARIF BAJWA) abstract                                    |
| X        | 1-11, 15          | GB2411086 A  
(VODAFONE) whole document                                          |
| X        | 1-11, 14          | EP1372298 A1  
(TELIASONERAFINLAND) whole document                                 |
| X        | 1-3, 15           | WO03/073304 A1  
(NOKIA) abstract                                                    |
| X        | 1-3, 15           | EP1783958 A1  
(SAMSUNG) whole document                                              |
| X        | 1-3, 15           | US2005/0266798 A1  
(MOLONEY) whole document                                               |
| X        | 1-3, 15           | GB2412819 A  
(NOKIA) whole document                                               |

**Categories:**

| X | Document indicating lack of novelty or inventive step |
| Y | Document indicating lack of inventive step if combined with one or more other documents of same category |
| & | Member of the same patent family |
| A | Document indicating technological background and/or state of the art |
| P | Document published on or after the declared priority date but before the filing date of this invention |
| E | Patent document published on or after, but with priority date earlier than, the filing date of this application |

**Field of Search:**

Search of GB, EP, WO & US patent documents classified in the following areas of the UKPC:

- H4L

Worldwide search of patent documents classified in the following areas of the IPC:

- H04B; H04L; H04Q

The following online and other databases have been used in the preparation of this search report.
### International Classification:

<table>
<thead>
<tr>
<th>Subclass</th>
<th>Subgroup</th>
<th>Valid From</th>
</tr>
</thead>
<tbody>
<tr>
<td>H04L</td>
<td>0012/28</td>
<td>01/01/2006</td>
</tr>
<tr>
<td>H04L</td>
<td>0029/06</td>
<td>01/01/2006</td>
</tr>
<tr>
<td>H04Q</td>
<td>0007/38</td>
<td>01/01/2006</td>
</tr>
</tbody>
</table>