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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
7 August 2014 (07.08.2014)

WIPO | PCT

(10) International Publication Number
WO 2014/121294 A1

(51) International Patent Classification:
G06Q 10/00 (2012.01)

(21) International Application Number:
PCT/US2014/014734

(22) International Filing Date:
4 February 2014 (04.02.2014)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/760,320 4 February 2013 (04.02.2013) US
61/913,859 9 December 2013 (09.12.2013) US

(71) Applicant: **SHOPKICK, INC.** [US/US]; 999 Main St., 2nd Floor, Redwood City, CA 94063 (US).

(72) Inventors: **EMIGH, Aaron, T.**; 999 Main St., 2nd Floor, Redwood City, CA 94063 (US). **SHRIVASTAVA, Vivek**; 999 Main St., 2nd Floor, Redwood City, CA 94063 (US). **STANEK, Steven**; 999 Main St., 2nd Floor, Redwood City, CA 94063 (US).

(74) Agent: **PUGALIA, Amita**; Van Pelt, Yi & James LLP, 10050 N. Foothill Blvd., Suite 200, Cupertino, CA 95014 (US).

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

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

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

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

(54) Title: PRESENCE DETECTION USING BLUETOOTH AND HYBRID-MODE TRANSMITTERS

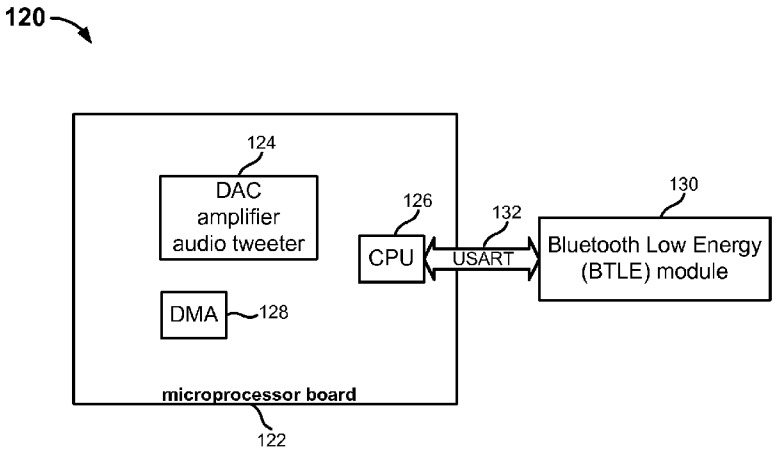

FIG. 1B

(57) Abstract: Presence detection using Bluetooth and hybrid-mode transmitters is disclosed. In some embodiments, one or more transmitters are configured to transmit an iBeacon broadcast and a proprietary Bluetooth Low Energy (BTLE) broadcast, wherein at least one of the transmitted broadcasts includes an identifier that specifies a venue. The broadcasts are captured by a handset and de-coded to infer presence of the handset at the venue.

WO 2014/121294 A1

PRESENCE DETECTION USING BLUETOOTH AND HYBRID-MODE TRANSMITTERS

CROSS REFERENCE TO OTHER APPLICATIONS

[0001] This application claims priority to U.S. Provisional Patent Application No. 61/760,320 entitled PRESENCE DETECTION USING BLUETOOTH AND HYBRID-MODE TRANSMITTERS filed February 4, 2013 which is incorporated herein by reference for all purposes. This application also claims priority to U.S. Provisional Patent Application No. 61/913,859 entitled PRESENCE DETECTION USING BLUETOOTH AND HYBRID-MODE TRANSMITTERS filed December 9, 2013 which is incorporated herein by reference for all purposes.

BACKGROUND OF THE INVENTION

[0002] Current technologies for detecting presence suffer from many limitations, including high costs of hardware and/or installation (e.g., for wifi triangulation) and reliance on handset audio channels that may vary (e.g., for ultrasonic audio based technologies).

BRIEF DESCRIPTION OF THE DRAWINGS

[0003] Various embodiments of the invention are disclosed in the following detailed description and the accompanying drawings.

[0004] Figure 1A is a high level block diagram illustrating an embodiment of components that may be employed for presence detection.

[0005] Figure 1B is a high level block diagram illustrating an embodiment of a transmitter.

[0006] Figure 2 is a flow chart illustrating an embodiment of a process for broadcasting a secure identifier.

[0007] Figures 3A-3B are flow charts illustrating embodiments of processes for validating a received secure identifier.

[0008] Figure 4 comprises flow charts illustrating embodiments of processes associated with updating the firmware on a transmitter.

[0009] Figure 5 comprises flow charts illustrating embodiments of processes associated with updating the transmitter identifier of a transmitter.

DETAILED DESCRIPTION

[0010] The invention can be implemented in numerous ways, including as a process, an apparatus, a system, a composition of matter, and/or a computer readable medium such as a computer readable storage medium or a computer network wherein program instructions are sent over optical or electronic communication links. In this specification, these implementations, or any other form that the invention may take, may be referred to as techniques. In general, the order of the steps of disclosed processes may be altered within the scope of the invention.

[0011] A detailed description of one or more embodiments of the invention is provided below. The invention is described in connection with such embodiments, but the invention is not limited to any embodiment. The scope of the invention is not limited by these embodiments, and the invention encompasses numerous alternatives, modifications, and equivalents. Numerous specific details are set forth in the following description in order to provide a thorough understanding of the invention. These details are provided for the purpose of example, and the invention may be practiced without some or all of these specific details. For the purpose of clarity, technical material that is known in the technical fields related to the invention has not been described in detail so that the invention is not unnecessarily obscured.

[0012] Presence detection, e.g., at a venue such as a store, can be used for a variety of purposes, such as presenting content relevant to the venue at which presence is detected (for example content that a user has indicated as favorite content or content that is recommended by a recommendation system using prior user behavior and/or externally available data such as Facebook Open Graph data as inputs) or issuing a reward in exchange for presence at the venue. Some benefits of presence detection are delineated in prior U.S. Patent Application No. 13/197,763, entitled METHOD AND SYSTEM FOR PRESENCE DETECTION, filed August 3, 2011, which is herein incorporated by reference in its entirety for all purposes, as if set forth in full herein. All applications enumerated therein may be used with respect to presence detection techniques as described herein, and its description of an ultrasonic-based transmitter can be taken as a description of possible operation of the audio-based transmitter component(s) described herein.

[0013] Figure 1A is a high level block diagram illustrating an embodiment of components that may be employed for presence detection. In the given example, handset 100 detects presence

of transmitter 102 on the premises of venue 104 and reacts accordingly, e.g., by communicating with server 106 via a wireless network. Transmitter 102 may be configured for one or more of the following functionalities: to broadcast store and department identifiers detectable by mobile users using both audio and Bluetooth technologies; to prevent replay attacks and rogue transmitters from cheating; to passively detect when a user enters a store and notify the user of any relevant information, including, but not limited to, notifying the user to open an application and redeem existing rewards (e.g., points); and to update the store and department identifiers broadcast by the transmitter remotely and securely from a mobile handset.

[0014] The term "handset" as used herein may refer to any mobile computing device, such as a smartphone, a tablet, etc. A typical handset, such as handset 100 of Figure 1A, includes a microprocessor, a memory coupled to the microprocessor providing both code that the microprocessor executes and data under the control of the microprocessor, a display (e.g., a touch-sensitive display) through which the user may view and interact with user interface elements, a Bluetooth transceiver, a microphone, a cellular and/or wifi transceiver, and various other components. For the purposes of this exposition, when it is said that a handset performs a technique, this should be taken to mean that the microprocessor performs operations according to the code stored in its associated memory and interacts with other components within the handset, as necessary, to perform said technique.

[0015] In some embodiments, a transmitter, such as transmitter 102 of Figure 1A, incorporates two technologies: a Bluetooth Low Energy (BTLE) module and an on-board audio module (e.g., comprising a DAC and tweeter) that allow the transmitter to broadcast identifiers in two media. Bluetooth and BTLE as used herein, as well as many Bluetooth specific terms known to those skilled in the art, are as set forth in the Specification of the Bluetooth System, version 4.0, dated June 30, 2010, available online from the Bluetooth Special Interest Group at https://www.bluetooth.org/Technical/Specifications/adopted.htm and/or https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=229737, which are herein incorporated by reference in their entirety for all purposes, as if set forth in full herein.

[0016] Figure 1B is a high level block diagram illustrating an embodiment of a transmitter. For example, transmitter 120 of Figure 1B comprises transmitter 102 of Figure 1A. As depicted, transmitter 120 includes a microprocessor board 122. Note that use of the terminology "board" does not imply that a separate physical board is needed; all components (or all components except the tweeter) may be included in a single physical board. Microprocessor board 122 comprises an onboard digital-to-analog converter (DAC) and broadcasts store and/or department identifier(s)

through an amplifier connected to a tweeter as depicted via block 124. As used herein, the term "tweeter" refers to a transducer capable of reproducing sound in an appropriate spectrum, e.g., ultrasonic frequencies relevant to an ultrasonic transmitter. As depicted, microprocessor board 122 further includes central processing unit (CPU) 126 and direct memory access (DMA) 128. In some embodiments, the DAC may be integrated into the CPU. Transmitter 120 further includes a Bluetooth Low Energy (BTLE) module 130. In some embodiments, BTLE module 130 comprises a standalone BTLE module that runs a minimalist stack and talks to microprocessor board 122 over universal synchronous/asynchronous receiver/transmitter (USART) link 132.

[0017] In some embodiments, a transmitter (e.g., transmitter 102 of Figure 1A or transmitter 120 of Figure 1B) broadcasts store and/or department identifier(s) using both audio and Bluetooth channels. In some embodiments, for audio, an identifier is encoded using ultrasonic frequencies and broadcast using a tweeter. In some embodiments, for Bluetooth, an identifier is embedded in the advertisement parameters of a Bluetooth beacon and broadcast as a service that can be detected by mobile handsets using passive scanning of Bluetooth channels.

[0018] Figure 2 is a flow chart illustrating an embodiment of a process for broadcasting a secure identifier or code. In some embodiments, code rotation is employed for preventing fraud. Process 200 specifically employs a nonce and hashing mechanism for preventing replay attacks. At step 202, a nonce is generated. A nonce, which is never repeated, may be periodically generated by a transmitter (e.g., transmitter 102 of Figure 1A or transmitter 120 of Figure 1B) using any appropriate technique. For example, in some embodiments, a hash of a current timestamp from the microprocessor or Bluetooth clock is used to generate the nonce. In another example, a simple monotonically increasing sequence that is updated periodically is used to generate the nonce. In such cases, the periodicity of the increments may be selected appropriately and/or changed dynamically. An example of an increment is ten minutes.

[0019] At step 204, the nonce generated at step 202 is combined with store and/or department identifier(s) that need to be broadcasted, and an irreversible hash is computed on the combined result. The term "hash" is used informally for the purpose of this discussion and includes a keyed hash with a secret key in addition to any explicitly mentioned content being hashed. The hash may comprise any cryptographic hash function known to those skilled in the art, such as SHA-1, MD5, etc. An example of what may be meant as a "hash" is an HMAC using SHA-1. HMAC is described in the Internet Engineering Task Force (IETF) RFC 2104, available online from the IETF at http://tools.ietf.org/html/rfc2104, which is included herein by reference for all purposes, as if set forth herein in its entirety.

[0020] At step 206, a secret hash key is stored. For example, the secret hash key is burned into a memory, such as within transmitter microprocessor code, and is not exposed to any external parties.

[0021] At step 208, the hash result is appended at the end of the combination to generate a secure identifier for broadcasting according to the following formula:

[0022] secure identifier = (nonce + store identifier + hash(nonce + store identifier))

[0023] As used herein, the store identifier may include a department identifier, either as a separate component of the store identifier, or as a single identifier that is associated, for example in a database, with separate store and department identifiers. In one example of a generated secure identifier, four bytes are used for the nonce, eight bytes are used for the store identifier, and four bytes are used for the hash.

[0024] At step 210, the secure identifier generated at step 208 is broadcasted. In some embodiments, a broadcast is preceded by appropriately configuring BTLE module 130 to update service parameters, e.g., via USART 132. In some embodiments, the Bluetooth service identifier is set to a prescribed (e.g., the shopkick) service identifier. The service identifier is a constant (e.g., 128 bit) universally unique identifier (UUID) to allow passive scanning under a handset operating system, such as iOS or Android. In such cases, the secure identifier may be broadcasted as service parameters for the constant (shopkick) service. Thus, step 210 may include broadcasting the constant service UUID and the secure identifier as the service parameters via Bluetooth beacons and optionally modulating it (or a modified version, such as the store identifier instead of the service identifier, or a version without the department code) using ultrasonic frequencies.

[0025] A secure identifier broadcast by a transmitter (e.g., transmitter 102 of Figure 1A or transmitter 120 of Figure 1B) may be captured by mobile handsets of users through an audio channel by capturing sound on a microphone, by actively scanning Bluetooth advertisements, and/or by utilizing features of a handset operating system to passively scan. For example, a list of service UUIDs to listen for may be provided, and a notification may be received when any identifier from the list is detected.

[0026] Figures 3A-3B are flow charts illustrating embodiments of processes for validating a received secure identifier.

[0027] Process 300 of Figure 3A, for example, may be employed by a mobile handset. At step 302, a secure identifier is detected. At step 304, the secure identifier is authenticated. For example, the secure identifier may be sent to an associated (e.g., shopkick) server for authentication. In alternate embodiments, such processing may be performed on the handset. If the server is unavailable for any reason, the secure identifier may be cached and relayed to the server when it becomes reachable. In order to support such offline walk-ins, each nonce may be valid for a predetermined period of time (e.g., one hour). At step 306, it is determined whether the secure identifier is authenticated. If the secure identifier is authenticated, a valid response is received at step 308 in response to the authentication processing of step 304. If the secure identifier is not authenticated, it is concluded at step 310 that the secure identifier is not valid. That is, a valid response is not received at step 310 in response to the authentication processing of step 304, but instead, for example, no response or an indication that the secure identifier is fraudulent may be received at step 310.

[0028] Process 312 of Figure 3B, for example, may be employed server side, e.g., at a server associated with a mobile application that detects the secure identifier at a handset. At step 314, a request to authenticate a secure identifier is received from a mobile application. At step 316, a hash is computed, and the result is compared with the hash bytes included with the secure identifier. For example, at step 316, an irreversible hash may be computed on the (nonce + store identifier) combination along with the secret key, e.g. using an HMAC, and the result compared with the hash bytes on the secure identifier. At step 318, it is determined if the secure identifier is valid. If it is determined that the secure identifier is valid (e.g., because the hash matches and the nonce in the secure identifier is still valid), a valid response is sent to the client at step 320. If it is determined that the secure identifier is not valid, the request is discarded and/or logged as a fraudulent attempt at step 322, and a valid response is not sent to the client.

[0029] An advantage of integrating a Bluetooth module in the transmitter is the ability to interact with the transmitter from a mobile handset, including the ability to update some components of the transmitter code and other parameters from a smartphone.

[0030] Figures 4-5 are flow charts illustrating embodiments of processes for updating the store and/or department identifier(s) assigned to a transmitter.

[0031] Figure 4 comprises flow charts illustrating embodiments of processes associated with updating the firmware on a transmitter, which changes the entire code on the transmitter and reboots the transmitter with the new code. Since the store and/or department identifier(s) are part

of the transmitter firmware, they are also updated in the process. Process 400 of Figure 4 may comprise, for example, a server-side process. At step 402, new firmware is computed. In some cases, the new firmware is computed with a new wave file encoding the transmitter code. Since Bluetooth, and especially BTLE, is suited for small amounts of data transfer, piecewise updating is employed in some embodiments. Accordingly, the entire new firmware is divided into small pieces at step 404. At step 406, the pieces are assigned to clients, e.g., when they walk-in and detect a transmitter and/or depending on their walk-in history. Process 408 of Figure 4 comprises a client-side process. At step 410, a chunk or piece of the new firmware is received. At step 412, the received piece of new firmware is sent to the transmitter, e.g., so that a small value corresponding to the received piece of new firmware can be updated on the transmitter. Process 414 of Figure 4 comprises a transmitter-side process. At step 416, pieces comprising the new firmware are received, for example, from one or more clients. At step 418, a digital signature of the new firmware (e.g., an HMAC or another digital signature technique known to those skilled in the art, such as one using an asymmetric encryption such as RSA) is checked. At step 420, it is determined whether the digital signature check is successful. If the digital signature check is successful, the firmware is updated and the transmitter is restarted at step 422. If the digital signature check fails, the received firmware is discarded at step 424, and the existing firmware of the transmitter is not updated.

[0032] Figure 5 comprises flow charts illustrating embodiments of processes associated with updating the transmitter identifier of a transmitter. In some embodiments, the transmitter identifier comprises a writable property (e.g., via Bluetooth GATT protocol) that can be updated by mobile handsets. Process 500 of Figure 5 comprises a server-side process. At step 502, a new identifier for the transmitter is signed with the key (e.g., using HMAC or another digital signature technique known to those skilled in the art, such as one using an asymmetric encryption such as RSA). At step 504, the signed copy is sent to a mobile client, e.g., when it reports that a particular transmitter has been detected which is deemed by the server to need updating (on demand), or depending on the client's walk-in history or geographic location, e.g., if the client has a history of walking into the venue containing the transmitter or is geographically close to it (prefetching). Process 506 of Figure 5 comprises a client-side process. At step 508, a signed identifier is received, e.g., at a mobile client from an associated server. At step 510, the signed identifier is sent to the transmitter for updating the identifier value. Process 512 of Figure 5 comprises a transmitter-side process. At step 514, a signed identifier is received, e.g., at a transmitter from a mobile client. At step 516, the signed identifier is validated using its key in a manner similar to that described above. At step 518, it is determined whether the identifier is valid. If it is determined that the

signed identifier is not valid, the transmitter identifier is not updated at step 520. However, if it is determined that the signed identifier is valid, the transmitter identifier is updated (for example, by communicating with the BTLE module via the USART), and the new identifier is broadcasted at step 522. Once the transmitter identifier is updated, a new audio wave file is generated on the transmitter using the new identifier. In some embodiments, process 506 optionally includes step 524 at which the handset verifies that the identifier has been updated and informs the server. In some embodiments, process 500 optionally includes step 526 at which an indication that the identifier has been updated is received at the server from the client and the database mappings of identifiers and venues/departments is accordingly updated at the server.

[0033] In some embodiments, Wake-on-Bluetooth is employed for enabling sound based detection. That is, Bluetooth may be used to trigger audio-based detection on mobile handsets. In this technique, Bluetooth scanning, as described above, is used to detect a nearby transmitter, which can then trigger audio detection in an associated application to figure out the exact venue that the user is in. Alternately, the store may be known from the Bluetooth identifier, but presence may not be inferred until a corresponding or compatible audio signal is detected.

[0034] In some embodiments, mobile phones are used as Bluetooth relays. Another key mechanism to upgrade existing transmitters without a Bluetooth module (i.e., which only support audio-based signaling) is to turn BTLE enabled handsets into mini ad-hoc transmitters when they are in a store fitted with an audio transmitter. In this scheme, once a handset detects a code via audio signaling, it can then broadcast the same code over its BTLE transmitter and enable other users in the vicinity to take advantage of Bluetooth based detection or wake up as discussed above. This mechanism allows the upgrade of existing infrastructure with BTLE without any additional costs or hardware. It may also be used with initial BTLE detection to extend the range of a BTLE transmitter. In either case, an ad hoc transmitter may be configured to cease operation when a criterion has been reached, such as a threshold period of time since the primary transmitter was detected or detection of a distance traveled (e.g., by accelerometer or GPS or other means on the handset) since the primary transmitter was detected.

[0035] In some embodiments, Bluetooth enabled transmitters may be used for fine-grained indoor mapping. That is, the signal propagation properties of Bluetooth may be leveraged to perform fine-grained location mapping including, for example, accurate identification of a user when walking into a store and tracking of the user as the user visits different sections in the store. Such functionality may be achieved, for example, by placing one or more Bluetooth enabled transmitters in every department of a store and recording the signal strength of both audio and

Bluetooth signals from all nearby transmitters on a mobile handset. In some such cases, clustering techniques (e.g., triangulation, sorting, etc.) may be employed to narrow the exact location of the mobile handset. Multiple signals from different transmitters can also be used as an authentication mechanism to safeguard against one-off transmitter thefts.

[0036] Many applications exist for the combined BTLE and audio-based transmitters described herein. As described, the addition of a BTLE module to the transmitter allows several tasks to be performed efficiently. For example, update of the store and/or department identifier(s) of a transmitter may be remotely accomplished. Moreover, a transmitter may be detected while an associated application is in the background using passive scanning of Bluetooth services as supported in iOS and Android. Such passive Bluetooth detection may trigger multiple actions depending on the situation including, for example, reminding a user to open an application to get rewards, popping up alerts about store and/or department specific offers, and using store specific information from an associated application and the user's profile to show targeted advertisements and reminders. Furthermore, a user may be suggested a store to check out when the user enters a mall. Since Bluetooth signals will bleed out of a store, they can be detected when the user is some distance away and could be used to suggest different stores based on latest promotions and the interests of the particular user.

[0037] Along with iOS 7, Apple introduced iBeacon. iBeacon comprises a specific format for broadcast messages that includes an identifier for the entity broadcasting as well as "major" and "minor" identifiers, which are capable of being used, for example, as store and/or department identifier(s). The exact format of an iBeacon broadcast can be readily observed by one skilled in the art by capturing the BTLE broadcasts of an iBeacon with known "major" and "minor" identifiers. An advantage of iBeacons is that iOS 7 includes OS-level support for iBeacons so that applications can register for being waken up on detection of an iBeacon with a specific identifier. However, a downside of iBeacons is that the standard iBeacon implementation makes it very easy to free ride on infrastructure, e.g., by registering to get alerted to a particular entity's beacons and then using the "major" and "minor" identifiers for location identification. If the iBeacon protocol is used in its basic form, these beacons can be used to wake up an application. Moreover, there is no code rotation capability, and it is trivial to forge any transmission, so iBeacon broadcasts are open to fraud.

[0038] In some embodiments, iBeacon and proprietary BTLE broadcasts are interleaved. That is, in order to take advantage of the wake-up feature of iBeacon, but to avoid free-riding by competitors, iBeacon advertisement packets are interleaved with custom proprietary broadcasts.

iBeacon signals just contain a 128 bit UUID, with nonsense major and minor identifiers. This is purposefully done to prevent competitors from using proprietary infrastructure for presence detection. As described in detail above, store and/or department information in proprietary broadcasts is encrypted and/or digitally signed. In one example, iBeacon signals are interleaved with proprietary signals in a ratio of 3:1 (this value is tuned to minimize the latency of waking up but at the same time minimizing the time taken to decode the correct store and/or department information from proprietary broadcasts), though other ratios are also workable.

[0039] In some embodiments, the major and/or minor iBeacon identifier(s) are rotated to prompt frequent application wake ups. In order to register for wake up, an application needs to specify either the iBeacon identifier or a combination of the iBeacon identifier and major and/or minor identifier(s). In the former case, iOS will only wake up an application once on entering a range of an iBeacon source that is broadcasting a desired iBeacon identifier. Hence even if multiple beacons broadcasting an iBeacon identifier exist, only the beacon that is encountered first by a device will wake up an application. The rest of the beacons do not lead to application wake ups. This can be a problem in larger deployments where it is desirable for an application to wake up to different beacons broadcasting the same iBeacon UUID. There exist multiple ways of solving this problem. One solution includes assigning different major and/or minor identifier(s) to every physical beacon and having an application then register to be woken up to all those combinations of iBeacon identifier and major and/or minor identifier(s). In this case, though, any third party application can identify the specific locations of all the beacons and can free ride on the beacon deployment. Thus, this solution or scheme may not be desirable. A more viable solution includes having every physical beacon broadcast the iBeacon identifier and rotating between a set of major and/or minor identifier(s). The application still registers to be woken up by all the set of major and/or minor identifiers, but there is no way for a competitor to map a particular beacon source to a major and/or minor identifier. The rotation scheme may be chosen such that probabilistically nearby beacons broadcast different major and/or minor identifiers, allowing an application to be woken up when it comes near different physical beacons.

[0040] In some embodiments, the iBeacon identifier is changed to prevent third party wake ups. In order to completely prevent third party applications from registering for wake ups from a prescribed, proprietary iBeacon identifier, the iBeacon identifier is updated from a client. Such an update can take place in phases. Until all the beacons are updated to the new iBeacon identifier, an associated application registers to be woken up by both the old and the new iBeacon identifiers, but once the update is complete, the application can switch to just the new iBeacon identifier.

Depending on the frequency of this update, it can make any third party applications that are unaware of the new iBeacon identifier unreliable for wake ups. Thus, any commercial application that relies on a prescribed iBeacon identifier to be constant will experience poor wake ups and will need to constantly monitor and/or sniff Bluetooth signals in the field to be abreast of the latest iBeacon identifier.

[0041] In some embodiments, the iBeacon broadcast may be completely disabled for any specific periods of time. That is, iBeacon support can be turned off completely if desired so that third party applications cannot use proprietary beacons at all for wake ups. This can be enforced for any random period of time or can be permanent. It provides a final defense for private infrastructure deployment against any form of free riding.

[0042] Although the foregoing embodiments have been described in some detail for purposes of clarity of understanding, the invention is not limited to the details provided. There are many alternative ways of implementing the invention. The disclosed embodiments are illustrative and not restrictive.

CLAIMS

1. A method, comprising:

generating an identifier that specifies a venue; and

configuring one or more transmitters to transmit an iBeacon broadcast and a proprietary Bluetooth Low Energy (BTLE) broadcast, wherein at least one of the transmitted broadcasts includes the identifier;

wherein the transmitted broadcasts are captured by a handset and decoded to infer presence of the handset at the venue.

2. The method of claim 1, wherein the identifier comprises a store identifier that specifies a store.

3. The method of claim 1, wherein the identifier comprises a department identifier that specifies a department in a store.

4. The method of claim 1, wherein the iBeacon broadcast is employed to wake up a handset application configured to detect an associated service identifier.

5. The method of claim 1, further comprising interleaving the iBeacon broadcast and the proprietary BTLE broadcast.

6. The method of claim 5, wherein iBeacon signals are interleaved with proprietary BTLE signals in a ratio of 3:1.

7. The method of claim 1, wherein the proprietary BTLE broadcast is encrypted.

8. The method of claim 1, wherein the proprietary BTLE broadcast is digitally signed.

9. The method of claim 1, further comprising rotating one or both of major and minor iBeacon identifiers to prompt frequent application wake ups.

10. The method of claim 1, further comprising changing the iBeacon identifier to prevent third party wake ups.

11. The method of claim 10, wherein the iBeacon identifier is updated via the handset.

12. The method of claim 1, wherein the identifier comprises a secure identifier.

13. The method of claim 1, wherein the identifier comprises a nonce.

14. The method of claim 1, wherein generating the identifier comprises computing a hash, wherein the hash includes at least one of a nonce value and a secret key.

15. The method of claim 1, further comprising configuring an ultrasonic transmitter to transmit the identifier via an ultrasonic channel.

16. The method of claim 1, further comprising updating at least one of the transmitters based on a directive received via the handset.

17. The method of claim 1, wherein the broadcasts captured at the handset are processed with respect to an application of the handset.

18. The method of claim 1, further comprising disabling the iBeacon broadcast for a prescribed period of time.

19. A system, comprising:

a processor configured to generate an identifier that specifies a venue; and

one or more transmitters configured to transmit an iBeacon broadcast and a proprietary Bluetooth Low Energy (BTLE) broadcast, wherein at least one of the transmitted broadcasts includes the identifier;

wherein the transmitted broadcasts are captured by a handset and decoded to infer presence of the handset at the venue.

20. A computer program product, the computer program product being embodied in a non-transitory computer readable storage medium and comprising computer instructions for:

generating an identifier that specifies a venue; and

configuring one or more transmitters to transmit an iBeacon broadcast and a proprietary Bluetooth Low Energy (BTLE) broadcast, wherein at least one of the transmitted broadcasts includes the identifier;

wherein the transmitted broadcasts are captured by a handset and decoded to infer presence of the handset at the venue.

21. A method, comprising:

generating a secure identifier, wherein the secure identifier comprises a venue identifier; and

configuring a transmitter to broadcast the secure identifier using Bluetooth Low Energy (BTLE);

wherein the secure identifier broadcast by the transmitter is captured by a handset and decoded to infer presence of the handset at a venue identified by the venue identifier.

22. The method of claim 21, wherein the venue identifier comprises one or both of a store identifier and a department identifier.

23. The method of claim 21, wherein the secure identifier includes a nonce.

24. The method of claim 21, wherein generating the secure identifier comprises computing a hash, wherein the hash includes at least one of a nonce value and a secret key.

25. The method of claim 21, further comprising configuring the transmitter to broadcast the secure identifier via an audio channel comprising ultrasonic frequencies.

26. The method of claim 21, further comprising configuring the transmitter to update either or both of the secure identifier and the transmitter firmware based on a directive received via the handset.

27. A method, comprising:

configuring a handset to scan via Bluetooth Low Energy (BTLE) to detect a prescribed service identifier and a corresponding service parameter;

decoding the service parameter, wherein decoding the service parameter includes extracting a venue identifier; and

inferring presence at a venue corresponding to the venue identifier.

28. The method of claim 27, wherein the service identifier comprises a constant universally unique identifier (UUID).

29. The method of claim 27, wherein the service parameter comprises a secure identifier generated by computing a hash including at least one of a nonce value and a secret key.

30. The method of claim 27, wherein the venue identifier comprises one or both of a store identifier and a department identifier.

31. The method of claim 27, wherein the handset is configured to actively scan Bluetooth advertisements.

32. The method of claim 27, wherein the handset is configured to passively scan via BTLE.

33. The method of claim 27, further comprising configuring the handset to receive a secure identifier comprising the venue identifier via an ultrasonic audio channel.

34. The method of claim 27, further comprising authenticating the service parameter.

35. The method of claim 27, further comprising receiving a reward in response to presence at the venue.

36. The method of claim 27, further comprising configuring the handset to update a transmitter that broadcasts the prescribed service identifier and corresponding service parameter.

37. The method of claim 27, further comprising configuring the handset to use BTLE detection to trigger audio-based detection.

38. The method of claim 27, further comprising configuring the handset as a relay to transmit the service identifier and service parameter via BTLE.

39. A system, comprising:

a transceiver configured to scan via Bluetooth Low Energy (BTLE) to detect a prescribed service identifier and a corresponding service parameter; and

a processor configured to decode the service parameter by extracting a venue identifier and infer presence at a venue corresponding to the venue identifier.

40. A computer program product embodied in a non-transitory computer readable storage medium and comprising computer instructions for:

configuring a handset to scan via Bluetooth Low Energy (BTLE) to detect a prescribed service identifier and a corresponding service parameter;

decoding the service parameter, wherein decoding the service parameter includes extracting a venue identifier; and

inferring presence at a venue corresponding to the venue identifier.

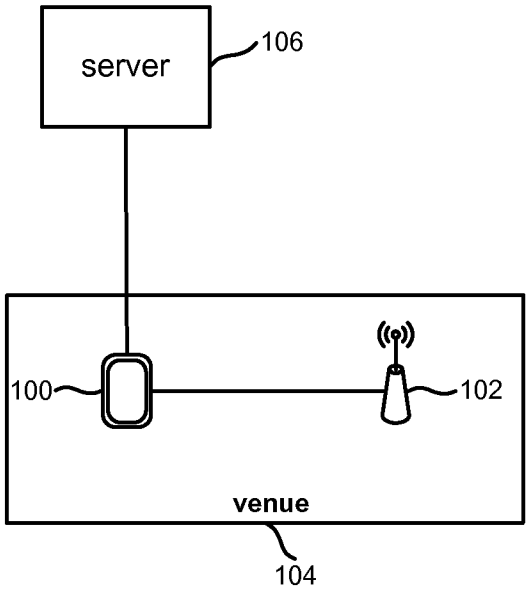

FIG. 1A

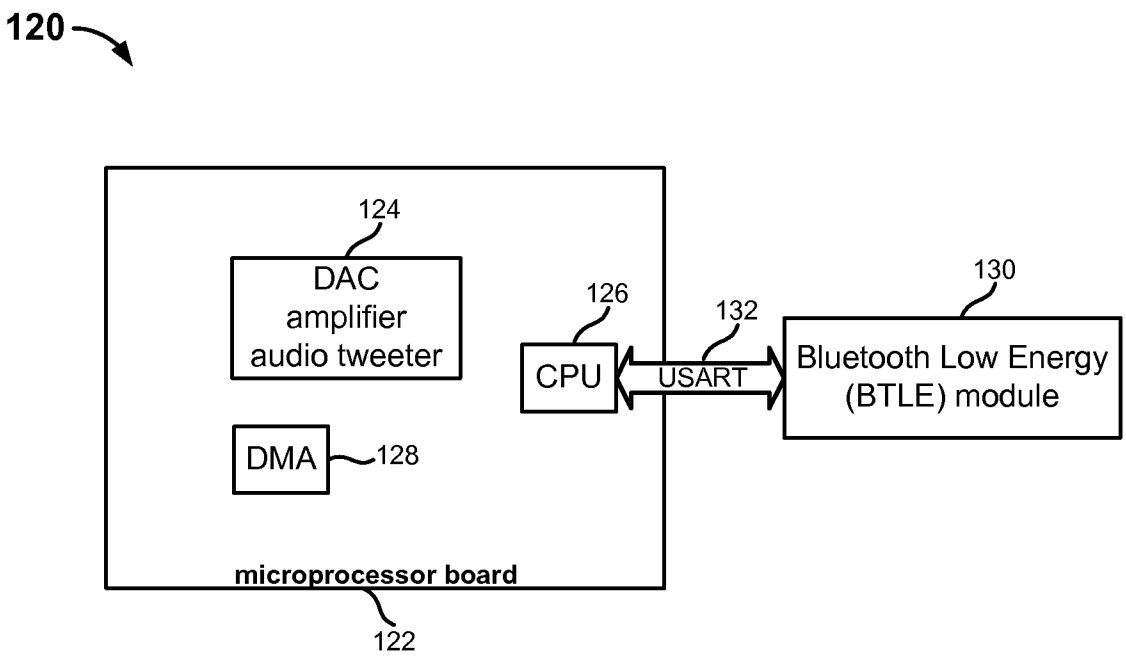
120
124
DAC
amplifier
audio tweeter
126
CPU
132
USART
130
Bluetooth Low Energy
(BTLE) module
DMA
128
microprocessor board
122

FIG. 1B

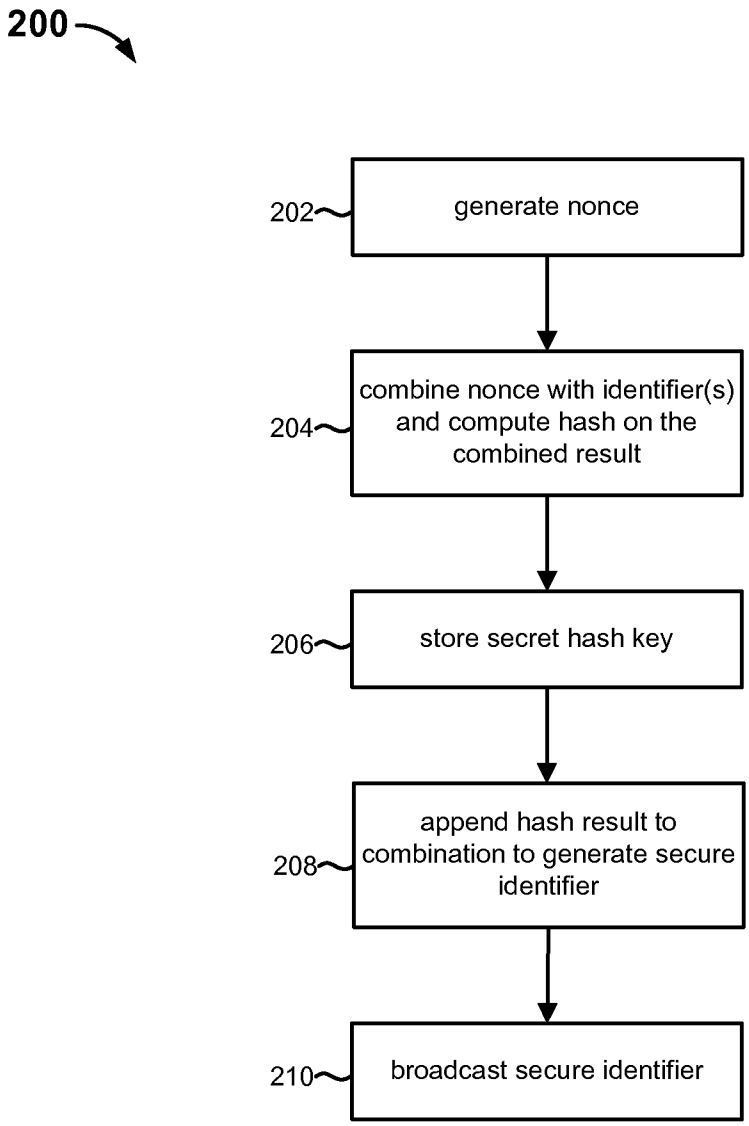
200
202
generate nonce
204
combine nonce with identifier(s) and compute hash on the combined result
206
store secret hash key
208
append hash result to combination to generate secure identifier
210
broadcast secure identifier

FIG. 2

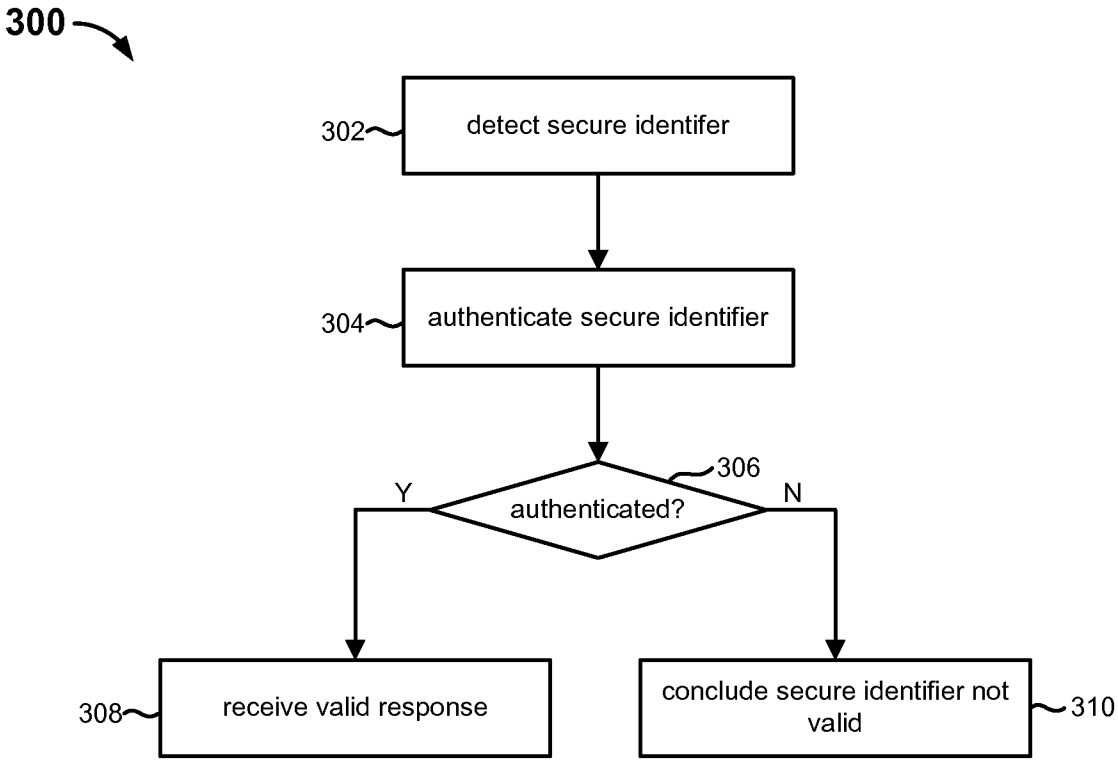
300
302
detect secure identifer
304
authenticate secure identifier
306
authenticated?
Y
N
308
receive valid response
conclude secure identifier not valid
310

FIG. 3A

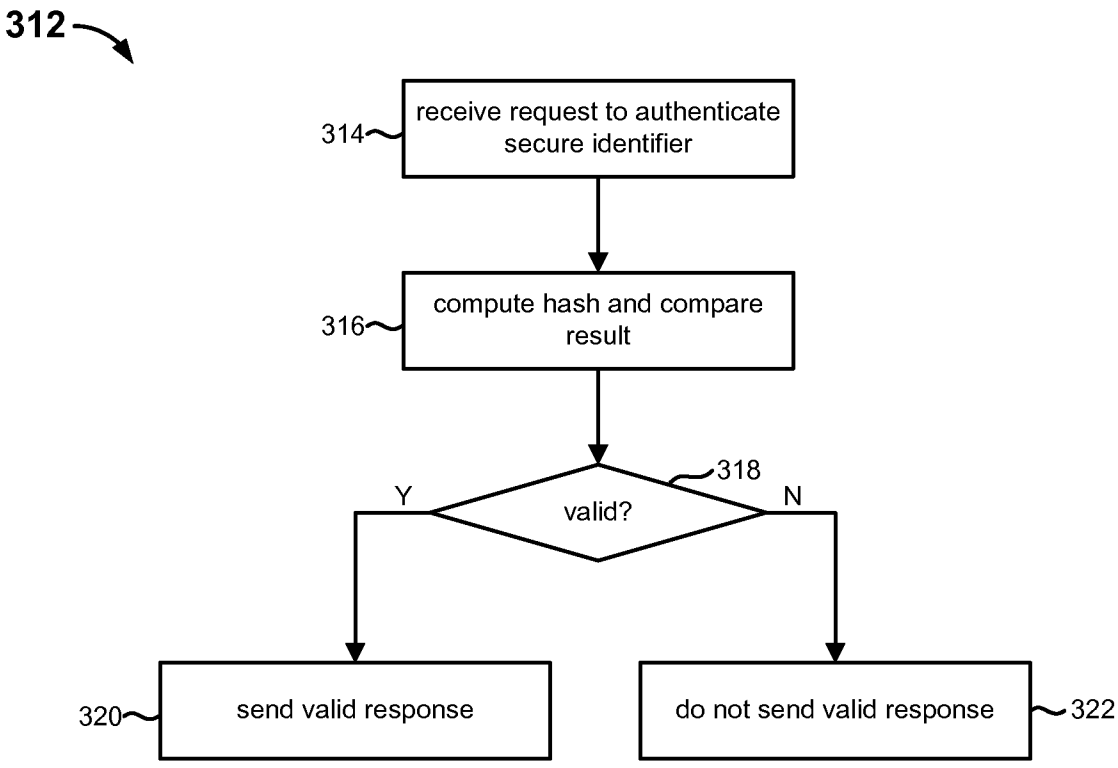
312
314
receive request to authenticate secure identifier
316
compute hash and compare result
318
valid?
Y
N
320
send valid response
do not send valid response
322

FIG. 3B

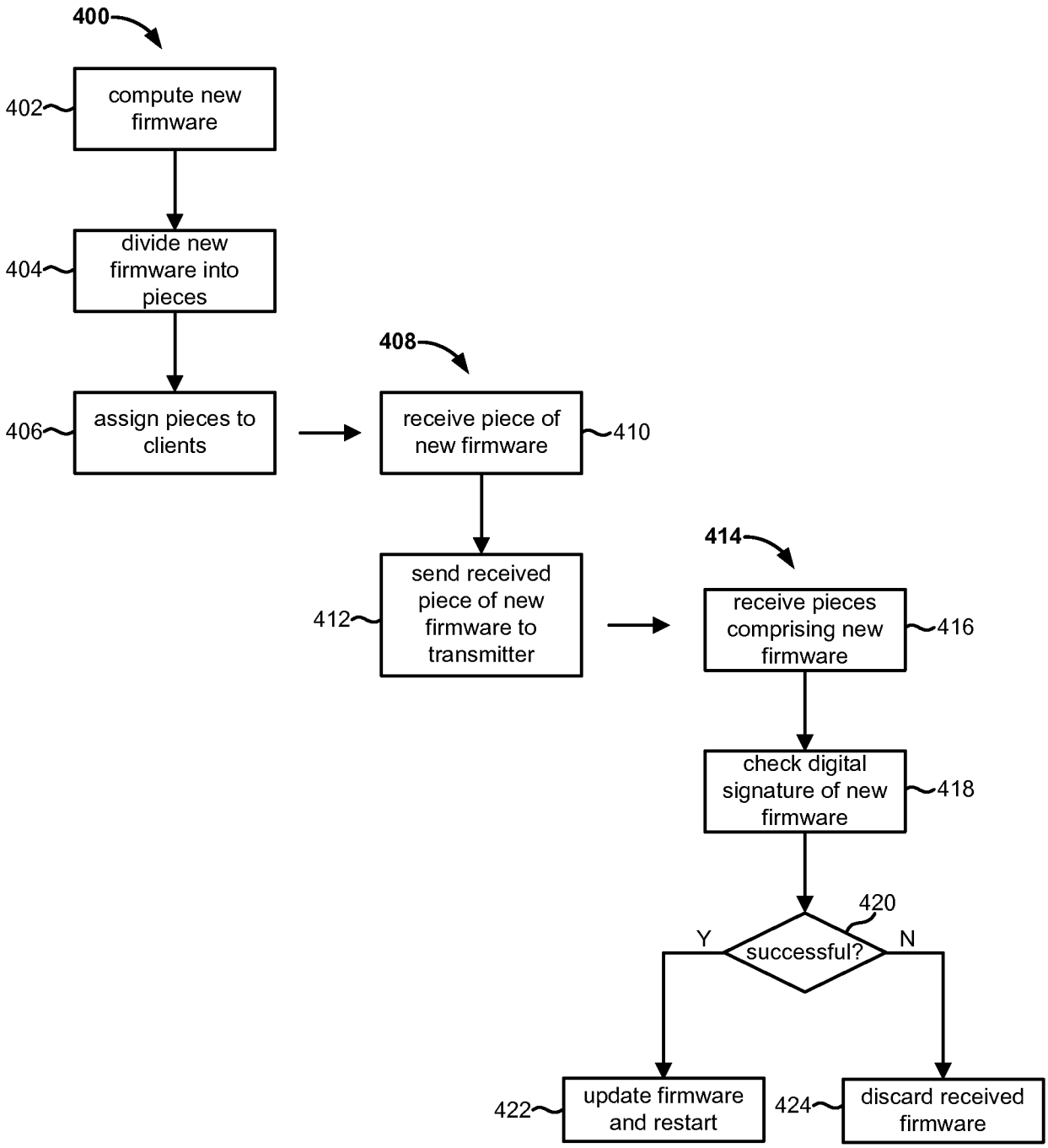
400
compute new firmware
402
divide new firmware into pieces
404
assign pieces to clients
406
408
receive piece of new firmware
410
send received piece of new firmware to transmitter
412
414
receive pieces comprising new firmware
416
check digital signature of new firmware
418
420
successful?
Y
N
update firmware and restart
422
discard received firmware
424

FIG. 4

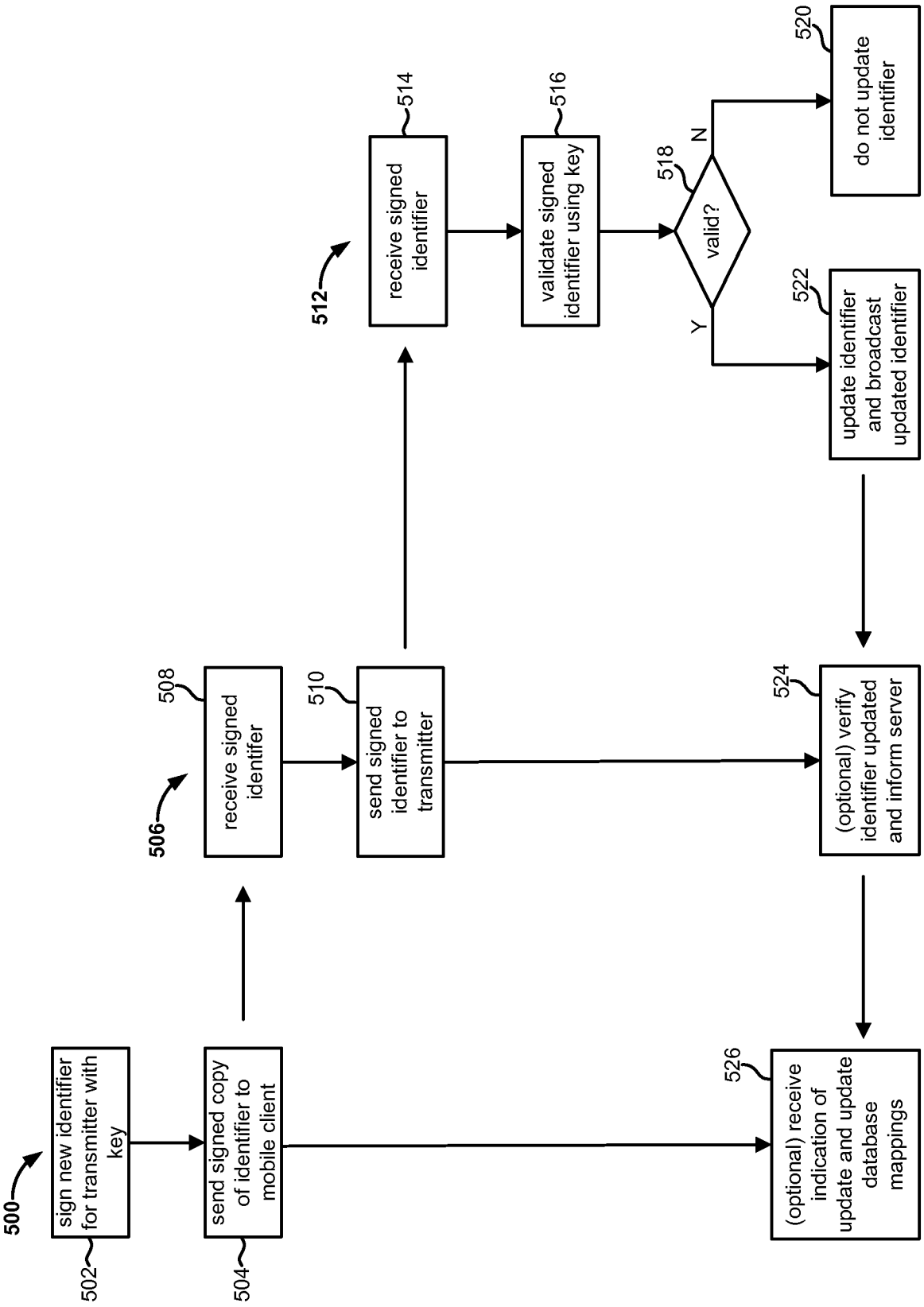
500
sign new identifier for transmitter with key
502
send signed copy of identifier to mobile client
504
receive signed identifer
506
508
send signed identifier to transmitter
510
receive signed identifier
512
514
validate signed identifier using key
516
valid?
518
N
Y
do not update identifier
520
update identifier and broadcast updated identifier
522
(optional) verify identifier updated and inform server
524
(optional) receive indication of update and update database mappings
526

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/014734

A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - G06Q 10/00 (2014.01)
USPC - 705/5
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC(8) - H04W 24/00; H04B 5/02; G06Q 10/00, 40/00 (2014.01)
USPC - 455/426.1; 705/1.1, 5, 14.49, 26.1

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
CPC - G 06Q 10/02; G 06Q 10/025 (2014.02)

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

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y -- A	US 2011/0029370 A1 (ROEDING et al) 03 February 2011 (03.02.2011) entire document	21-40 --- 1-20
Y	WO 2010/109271 A1 (IMMONEN) 30 September 2010 (30.09.2010) entire document	21-40
Y	US 2008/0040713 A1 (SUBBAKRISHNA et al) 14 February 2008 (14.02.2008) entire document	26, 36
Y -- A	US 2010/0063867 A1 (PROCTOR, JR. et al) 11 March 2010 (11.03.2010) entire document	21, 27 --- 1-20
X, P	Rao. PayPal Debuts Its Newest Hardware, Beacon, A Bluetooth LE Enabled Device For Hands-Free Check Ins And Payments. Techcrunch.com. 09 September 2013 (09.09.2013). Retrieved on [05-10-2014]. Retrieved from the Internet: URL <http://techcrunch.com/2013/09/09/paypal-debuts-its-newest-hardware-beacon-a-bluetooth-le-enabled-device-for-hands-free-check-ins-and-payments/> entire document	1-35
X, P	US 2013/0281084 A1 (BATADA et al) 24 October 2013 (24.10.2012) entire document	1-35

☒ Further documents are listed in the continuation of Box C. ☐

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
10 May 2014	02 JUN 2014
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-3201	Authorized officer: Blaine R. Copenheaver PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774

Form PCT/ISA/210 (second sheet) (July 2009)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/014734

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A, P	Lovejoy. iBeacon briefing: What is it, and what can we expect from it? 9to5Mac.com. 27 September 2013. Retrieved on 10 May 2014. Retrieved from the Internet: URL <http://9to5mac.com/2013/09/27/ibeacon-briefing-what-is-it-and-what-can-we-expect-from-it/?> entire document	1-20
L	Dilger. Inside iOS 7: iBeacons enhance apps' location awareness via Bluetooth LE. Appleinsider.com. 19 June 2013. Retrieved on [05-10-2014]. Retrieved from the Internet: URL <http://appleinsider.com/articles/13/06/19/inside-ios-7-ibeacons-enhance-apps-location-awareness-via-bluetooth-le> entire document	1-20
L	Betters. Apple's iBeacons explained: What it is and why it matters. Pocket-lint.com. 18 September 2013. Retrieved on [05-10-2014]. Retrieved from the Internet: URL< http://www.pocket-lint.com/news/123730-apple-s-ibeacons-explained-what-it-is-and-why-it-matters> entire document	1-20