Applications of Secure Location Sensing in Healthcare

Paul D. Martin
Harbor Labs
paul@harborlabs.com

Michael Rushanan
Harbor Labs
mike@harborlabs.com

Thomas Tantillo
Johns Hopkins University
tantillo@cs.jhu.edu

Christoph U. Lehmann
Vanderbilt University
christoph.u.lehmann
@vanderbilt.edu

Aviel D. Rubin
Johns Hopkins University
rubin@cs.jhu.edu

ABSTRACT

Secure location sensing has the potential to improve healthcare processes regarding security, efficiency, and safety. For example, enforcing close physical proximity to a patient when using a barcode medication administration system (BCMA) can mitigate the consequences of unsafe barcode scanning workarounds. We present Beacon+, a Bluetooth Low Energy (BLE) device that extends the design of Apple's popular iBeacon specification with unspoofable, temporal, and authenticated advertisements. Our prototype Beacon+ design enables secure location sensing applications such as real-time tracking of hospital assets (e.g., infusion pumps). We implement this exact real-time tracking system and use it as a foundation for a novel application that applies location-based restrictions on access control.

Categories and Subject Descriptors

D.4.6 [**Security and Protection**]: Access Controls

General Terms

Security

Keywords

BLE, location sensing, and beacon

1. INTRODUCTION

Tracking and managing assets in real-time are critical for large organizations such as Hospitals. For example, "more than [one-third] of nurses spend at least 1 hour per shift searching for equipment and the average hospital owns 35,000 inventory SKUs and utilization hovers around 32-48%, with nearly $4,000 of equipment per bed, lost or stolen each year" [8]. Moreover, tracking needs to be secure; specifically, it needs to be resilient to active and passive attacks that aid in the misappropriation of assets. We implement a real-time tracking system using low-cost Bluetooth Low Energy (BLE) devices that provide authenticated wireless communication to track securely assets and people.

We track assets in our system with an external device that can receive BLE transmissions containing location data[1], and send location data to a trusted server via Wi-Fi. We implement a device we call *Beacon+* to broadcast location data via BLE. This type of BLE beacon extends the design of Apple's popular iBeacon specification [17] by modifying the advertisement, or unidirectional broadcast, to contain a monotonically increasing sequence number and message authentication code (MAC).

In particular, the sequence number provides temporal freshness that is resilient to clock skew without synchronization. The MAC authenticates the Beacon+ to a trusted server, where the trusted server maintains the absolute location of each Beacon+. Upon receiving the Beacon+ advertisement, the server updates the location of an asset.

We use the real-time tracking system as a foundation for secure location sensing applications. One such example is access control that enforces location-based restrictions. This application relies on the authenticity of received Beacon+ advertisements to compute the relative location to an asset and provide access to asset data if and only if the accessor (i.e., the person who requires the data) is within close physical proximity. Location here is only one factor in a multi-factor access control scheme. For example, nurses and physicians who are away from their personal computer but moving around with a hospital-issued tablet must log in to the tablet with their credentials and be within close physical proximity of a patient to access her medical record.

Another secure location sensing application we describe is BCMA physical proximity enforcement. BCMAs typically involve scanning barcodes on patients and medications to interface with electronic records. Koppel et al. [31] identify 31 unique causes where healthcare professionals use workarounds to BCMA processes that they consider impractical (e.g., time). However, these workarounds can result in the wrong administration of medication which impacts patient safety. Therefore, physical proximity enforcement can integrate BLE receivers into scanning devices and require the user to be in an approved location to enable scanning.

The linchpin of our applications is Beacon+. To build a secure and interoperable Beacon+ device we require the following capabilities: (1) perform symmetric key opera-

[1]Assets that support BLE do not require an additional device.

BCB '16, October 02 - 05, 2016, Seattle, WA, USA

ACM 978-1-4503-4225-4/16/10...$15.00

DOI: http://dx.doi.org/10.1145/2975167.2975173

tions; (2) modify advertisement fields; (3) transmit unidirectional advertisements, and; (4) retain traditional beacon (e.g., iBeacon) advertisement structure. We are aware of only one similar, authenticated beacon called Trusted Beacon (TB) [20]. Beacon+ differs from TB in its choice of cryptographic primitive and number of advertisements for a single transmission. Specifically, TB lacks (1), (2) and (4).

Moreover, TB uses a weak, factorable [6, 18] 320-bit asymmetric RSA private to sign a random value that is valid for 5 minutes. An attacker can, therefore, replay a capture advertisement for up to 5 minutes. In contrast, Beacon+ uses a 128-bit symmetric AES key to compute a MAC on a monotonically increasing sequence number that is only valid for 1 second. Beacon+ conforms to the iBeacon standard because it fits in a single advertisement whereas TB requires multiple advertisements (i.e., the signature is longer than the message to be signed).

2. BACKGROUND

While prior work exists for the design of location-based access control protocols [33, 1, 30], there has, to the best of our knowledge, been little work done regarding their implementation and evaluation. Existing technologies such as RFID [25], GPS [21], and WiFi [8] have had varying levels of success on tracking and managing assets. In this section, we will explore the functionality of these technologies, and discuss how their limitations necessitated Beacon+.

2.1 Radio Frequency Identification

Radio Frequency Identification (RFID) provides short-range asset tracking using *tags* and *readers*. Readers interrogate tags and receive unique identifiers along with other data, and typically placed at ingress and egress points of a particular area [2]. The readers then read all tags entering or leaving the monitored area. Communication range for RFID is limited to tens of centimeters, and different bands of RFID communication (low frequency, high frequency, ultra high frequency) can increase the range up to 12 meters [11]. However, higher frequency RFID requires expensive antennas to extend the range. Deploying these antennas throughout an extensive area is impractical and can be considered unsafe depending on hospital RF safety policies.

2.2 Global Positioning System

Global Positioning System (GPS) is a reliable global satellite system for providing time and location information to any receiver with a clear view of at least four satellites. GPS is well-suited to outdoor tracking applications, but it does not function well when there is no direct line of sight to at least four satellites. Thus, GPS is not suitable for establishing indoor positioning [23] because it is often not accurate enough within buildings.

2.3 Wi-Fi

Wi-Fi facilitates wireless networking over mid-ranged distances. Multiple wireless access points are often used to provide coverage to large areas.These access points each have unique identifiers that bind to specific locations. Therefore, an administrator could track the location of individual clients by observing the order and location in which the clients connect with access points over a given period.

Wi-Fi meets the accuracy, timeliness, and communication range requirements for indoor position management and tracking. Previous work has looked at using Wi-Fi tags for exactly this purpose [34]. One of the benefits of Wi-Fi-based solutions is easy adoption; Wi-Fi tags are attached to devices or staff and communicate with existing access points. However, adhesive Wi-Fi tags are not securely integrated with the devices they manage as tags can be mixed up or maliciously removed. Also, Wi-Fi is not as power efficient as other technologies, it requires an additional layer of management (e.g., password, SSID, etc.), and it requires bidirectional communication that increases the attack surface. For example, an attacker can continuously communicate with the Wi-Fi device, attempting to authenticate and gain access.

2.4 Near Field Communication

Near field communication (NFC) [38] was invented for extremely short-range communication, on the order of several inches. Therefore, for the applications considered in this work, NFC is infeasible, as it would require an unreasonable number of NFC devices.

2.5 Bluetooth

Bluetooth [35] is a short-range communication protocol supported by most mobile devices (i.e., smartphones and laptops). Bluetooth-enabled devices initiate connections to host devices by entering *discoverable mode* and waiting for a scanning device to make a connection inquiry. The device then responds to the connection inquiry by sending information including a device name and a device class. If the host chooses to connect to the client device, then the two devices go through a *pairing process*.

Bluetooth technology has been used to build tracking systems [10, 4, 16, 27]. Previous work has generally used older Bluetooth versions (older than v4.0) and did not consider security as a design goal. Some tracking systems required tracked entities to establish connections with Bluetooth infrastructure devices resulting in two-way communication with potentially *untrusted entities* [27].

Beacons. Nokia introduced Bluetooth low energy (BLE) in 2004 as a wireless personal area network that later integrated into the Bluetooth 4.0 standard in 2010 [13]. BLE uses significantly less power than classic Bluetooth, and BLE devices can advertise information to a host device (*receiver* herein) without requiring the host device to pair. Conceptually, BLE is similar to NFC, but it is capable of operating at much longer ranges than NFC. In short, devices that need to broadcast small snippets of data at irregular intervals use BLE.

Beacon is one implementation of BLE. A beacon is an inexpensive BLE device (in the range of $5 [12] to $30 [9]) that repeatedly broadcasts a fixed unique identifier. Applications interpret these identifiers for a variety of purposes. For example, Apple's iBeacon [7] broadcasts what it calls an advertisement. The packet structure of an advertisement reveals a tuple of fixed identifiers that are interpreted by as coupon data.

Beacon+ bases itself on the iBeacon protocol and thus we adopt their advertisement structure. In particular, this structure is composed of the following fields [7]:

- UUID: a sixteen-byte unique number used to identify all iBeacons in a particular deployment.

- Major: a two-byte number used to identify groups of iBeacons within a deployment from other groups.
- Minor: a two-byte string used to identify individual iBeacons in a particular cluster of devices.

Although previous work has looked at using Beacons for indoor tracking [39, 5, 28, 19], the insecurity of the iBeacon protocol makes it poorly suited for this task in the presence of an attacker.

3. THREAT MODEL

We describe Beacon+ as having unspoofable, temporal and authenticated advertisements; as such, we recognize the following security goals unique to Beacon+.

1. *Integrity.* Advertisements should not be modifiable by an unauthorized entity.
2. *Availability.* Advertisements should be accessible.

We omit confidentiality because Beacon+ advertisements contain no private data. Moreover, we do not claim any privacy goals for Beacon+ as the application of tracking relinquishes the privacy of an asset or person inherently.

Attackers are distinguished based on their goals, capabilities, and relation to Beacon+. Thus, we have the following classification criteria.

1. *Active/Passive Attacker.* Active attackers can read, modify, and inject advertisements (i.e., BLE communication). Passive attackers can eavesdrop advertisements.
2. *Internal/External entity.* Internal entities have legitimate Beacon+ access (e.g., hospital administrator).
3. *Single/Coordinated group entities.*
4. *Sophisticated/Unsophisticated Attacker.* Sophisticated attackers have access to specialized equipment (e.g., high gain antennas). Unsophisticated attackers have access to conventional equipment (e.g., BLE sniffers).

An attacker may use Beacon+, the BLE device, smartphone, and the trusted server as *attack surfaces*. For example, an attacker may disrupt Beacon+ advertisements by physically destroying Beacon+ devices, or jamming or dropping advertisements. We classify Beacon+ security threats into the following categories:

1. *BLE interface threats.* An attacker can passively eavesdrop on advertisements, or actively jam, replay, modify, forge, or drop advertisements.
2. *Software threats.* An attacker can alter the logic of Beacon+ through software vulnerabilities.
3. *Application threats.* An attacker can compromise the intended functionality of an application.

Application-specific threats are unique to Beacon+ and non-obvious. For example, an active attacker may attempt to circumvent location-based restrictions by physically moving all Beacon+s to one central location. There exists threats to BLE devices, smartphones, and trusted servers that we do not cover because it is beyond the scope of Beacon+.

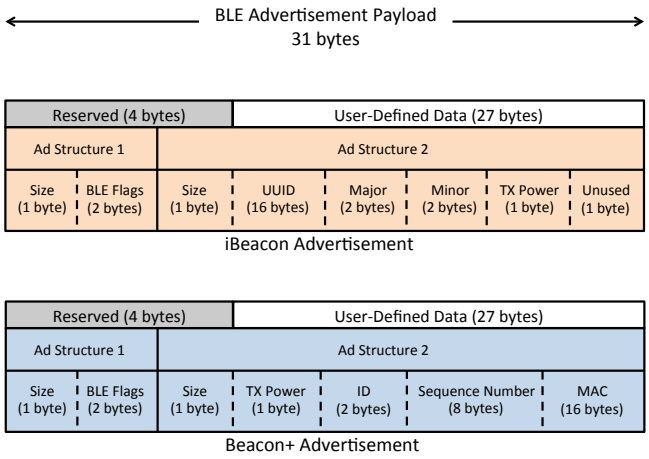

Figure 1: iBeacon and Beacon+ advertisement formats. BLE advertisements can support up to a 31-byte payload – 4 bytes are reserved for BLE structures and flags, leaving 27 bytes for user-defined data.

4. BEACON+

Apple's iBeacon and the majority of other beacons lack authentication and therefore are susceptible to spoofing; i.e., an attacker can advertise another beacon's UUID to trick receivers into believing that the beacon is within range. These beacons also lack a mechanism to provide receivers with a notion of time or, specifically, the notion of advertisement generation (temporal freshness).

Beacon+ prevents spoofing by adding lightweight authentication by way of a MAC, and it provides temporal freshness via a monotonically increasing sequence number. Each BLE advertisement has both MAC and sequence number appended to it. This advertisement maintains the single 27-byte payload structure and unidirectional broadcast protocol defined in the iBeacon specification [17].

Upon initialization, each Beacon+ is assigned a unique identification number that we distinguish from the UUID of regular beacons by labeling it as *ID*, an initial value for the monotonically increasing sequence number, and a secret key that is used to compute a MAC. The secret key is assigned a priori to deployment. As with current beacons, the TX Power (i.e., signal strength) to the Beacon+ at 1 meter in Decibel-milliwatts (dBm) is measured and set. The ID, current sequence number, secret key, and TX Power are stored in non-volatile memory on the Beacon+ to ensure that the values persist even if removing power.

The trusted server maintains both the initial sequence number and the secret key that will authenticate Beacon+ advertisements and check for temporal freshness. Beacon+ computes a MAC on the concatenation of TX Power, ID, and current sequence number with padding. Each second, Beacon+ increments it sequence number, computes a new MAC, and replaces the previous advertisement with the current one.

Figure 1 compares the advertisement format of Beacon+ and iBeacon. Beacon+ uses 2 bytes for the ID and 8 bytes for the monotonically increasing sequence number. One restriction of this specific byte allocation is that it supports only 216 or 65535 IDs. We choose to use 2 bytes for the ID

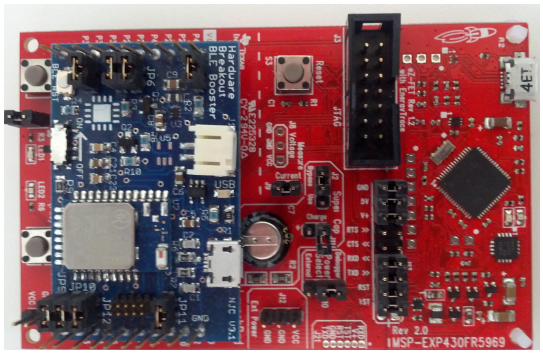

Figure 2: Beacon+ is implemented using the TI MSP430 LaunchPad (underlying red board) and Bluegiga Bluetooth BLE BoosterPack.

in order to allocate 8 bytes for the sequence number.

Beacon+ broadcasts advertisements at a predetermined rate. Faster rates (e.g., eight times per second) improve the likelihood that receivers detect Beacon+ devices in range but increase the power consumption. Slower rates conserve power consumption but may result in receivers failing to detect Beacon+s in range. We configure Beacon+ to broadcast advertisements at a rate of eight times per second (i.e., every 125µs) which matches the rate of iBeacon.

We represent time using monotonically increasing sequence numbers that increment at a regular timeout of once per second. The trusted server maintains the initial and subsequent sequence numbers, and upon receiving an authenticated advertisement, it will compare the received sequence number with the highest seen so far. The advertisement is accepted if the received number is not more than some threshold below the highest seen.

4.1 Implementation

We implemented the Beacon+ specification using the Texas Instruments MSP430FR5969 LaunchPad Development Kit [36] and Bluegiga Bluetooth Low Energy BoosterPack for the LaunchPad [15] (see Figure 2). The MSP430 board runs the control logic of Beacon+. During initialization, each MSP430 board is assigned an ID, starting sequence number (usually 1), secret key, and the appropriately calibrated TX Power. We place the MSP430 board at a chosen location in the environment, and we share the ID, starting sequence number, secret key, and chosen location with the trusted server.

Once per the timeout rate, the MSP430 board increments the sequence number, computes the MAC using AES-128 bit CBC-MAC, and sends the new advertisement to the BLE BoosterPack via the UART communication interface. The BLE BoosterPack receives the latest advertisement from the MSP430 and sends it out at a regular interval of eight times per second. The transmitted advertisements are then collected by devices moving throughout the environment and passed to the trusted server for validation (see Section 5).

5. APPLICATIONS

Beacon+ serves as a foundation for building many secure location sensing applications. We describe and implement two such applications, namely secure real-time asset tracking and location-based restrictions on access control. We also describe BCMA physical proximity enforcement.

5.1 Secure Real-Time Asset Tracking System

The tracking system is composed of three components: (1) Beacon+, (2) BLE–speaking devices that will be tracked (e.g. smartphone or tablet), and (3) backend server (*trusted server* hereon) that validates Beacon+ advertisements and calculates tracked devices' positions. The system is initialized by placing Beacon+s throughout the environment at chosen locations that provide good coverage of the area. This chosen location and the Beacon+'s assigned unique ID, secret key, and starting sequence number is shared with the trusted server, which is run by the system administrator [2]. As per the specification, each Beacon+ periodically broadcasts the authenticated BLE advertisement containing its unique ID, monotonically increasing sequence number, TX Power, and the corresponding MAC of the data.

Tracked BLE–speaking devices periodically collect the authenticated BLE advertisements and corresponding received signal strength (RSSI) from all Beacon+ within range. The device then sends a device update that contains the latest collected Beacon+ advertisements to the server using some other communication medium such as Wi-Fi, cellular, or wired LAN. This device functionality can be added to existing medical devices that support BLE with only a small modification, while older devices can use a BLE module or data collector (e.g., smartphone or computer).

To track personnel, each individual can carry their own smartphone or borrowed hospital-issued tablet. These types of computing devices are increasingly used in health-related environments due to the adoption of health information technology and Bring-Your-Own-Device (BYOD) [3]. An App is installed on the devices that collects Beacon+ advertisements and sends them over Wi-Fi or cellular networks to the trusted server.

Figure 3 shows an example of two different devices that are tracked. The first device is a physician's iPhone, which can communicate directly to the trusted server. The second device is a heart rate monitor that cannot communicate directly with the trusted server, and relies on a data collection computer to forward communication. In both cases, the devices collect the authenticated BLE advertisements from the Beacon+ within range, aggregate the advertisements and corresponding RSSI values, and send them to the backend server, which will use this information to determine the location of the device.

Upon receiving a device update, the trusted server validates each of the Beacon+ advertisements contained within that update. The trusted server checks, using the shared secret key for each Beacon+, that the MAC appended on an advertisement matches the computed MAC over the data. If the MAC does not match, that advertisement is discarded and not included in the location calculation. In addition, each advertisement is checked for freshness by comparing the monotonically increasing sequence number on the advertisement with the highest received sequence number received so far from that Beacon+. If the sequence number on the advertisement is not within a valid range of the highest sequence number seen to date (e.g., more than X sequence numbers older), that advertisement is not valid.

After Beacon+ advertisements in a device update are vali-

[2]Administer can return to a Beacon+ to refresh keys, apply firmware updates, or even replace it entirely.

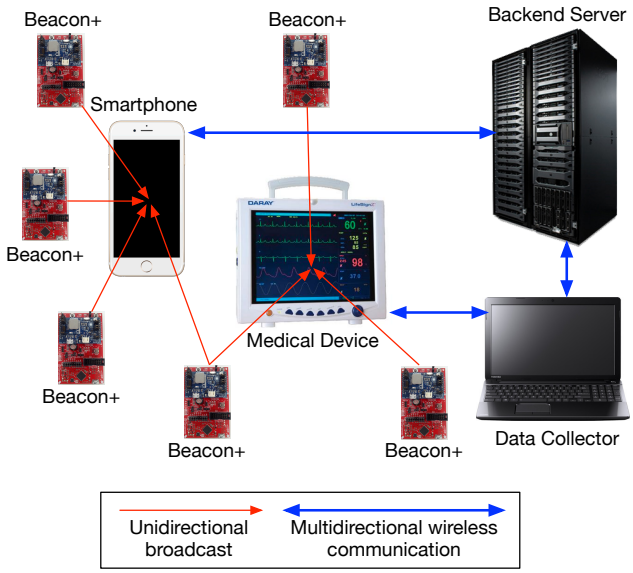

Figure 3: Secure Real-Time Asset Tracking System based on Beacon+.

dated, the trusted server can compute the location of the device. Given a device's RSSI value to a Beacon+, the trusted server can calculate the distance between the two entities using the following equation:

$$rssi = -10n * log_{10}(d) + A \quad (1)$$

$$d = 10^{\frac{(rssi-A)}{-10n}} \quad (2)$$

where $rssi$ is the measured received signal strength in dBm, A is the signal strength to the Beacon+ (in dBm) at 1 meter (i.e., the TX Power), d is distance in meters between the Beacon+ and the device, and n is the propagation constant or path-loss exponent (free space has $n = 2$ for reference, this value should be calibrated depending on the environment).

The trusted server can determine the location of the device using trilateration [22, 37, 24] given the distance calculation between the device and at least three Beacon+s and pre-existing knowledge of the physical location of each Beacon+. The device is located at the intersection of three circles, one circle centered at each Beacon+, where the radius of each circle is equal to the distance calculated between the device and that Beacon+. In order to track a device's position at all times using trilateration, it must be within range of at least three Beacon+ in order for the computation to succeed at the trusted server.

In addition to computing a device's location, the trusted server continually updates a database, which contains the location of each Beacon+, the location of each tracked device, acceptable boundaries for each device, and a log of system events. A web application reads the database an displays the location of each Beacon+ and tracked devices, the boundaries of each device, and the system events as they occur in real-time. The trusted server and web application can take action (e.g., raise an alarm, send an email or text message) in response to problematic events, such as when a device has left or is close to leaving the acceptable boundary.

Figure 5 shows a snapshot of an example web application

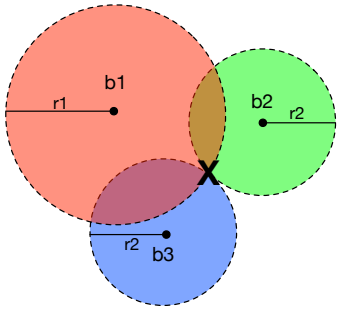

Figure 4: Trilateration Example. $r1$, $r2$, and $r3$ (radius of the b1, b2, and b3 circles respectively) correspond to the calculated distance between the tracked device and each Beacon+. The intersection of the three circles (marked by an X) determines the location of the device.

that visualizes the location of 10 Beacon+ (blue circles), one device being tracked (solid red block), and the acceptable boundary of that device (red square outline) on a single floor of a university building. The web application enforces access control to ensure that the location of devices (and Beacon+) can only be seen by authorized individuals.

Attack Mitigations. An active attacker may steal a device. However, since devices are tracked in real-time, the appropriate authority is notified if the device moves outside its intended location. The attacker may also physically damage a Beacon+, remove the power source, or perform a sophisticated wireless jamming attack. The tracking system expects Beacon+ advertisements and device updates (i.e., heartbeat) at regular intervals; therefore, the trusted server can implement a detection policy (much like a network intrusion detection system) that generates alerts. Or, the trusted server can generate audit logs for retroactive analysis.

5.2 Location–Based Restrictions

Sensitive data such as electronic medical records are protected using encryption and single-factor access control mechanisms (e.g., PIN numbers, passwords) to limit access to authorized individuals. However, this approach raises a major security concern as an attacker that is able to bypass or break the access control security gains access to all of the sensitive data in the database with a single breach. This threat is made worse in the context of a hospital, where computing devices are often used to access sensitive patient information, and a stolen or compromised device can provide an attacker with a large portion of private data.

To address this threat, we implement a prototype application that provides an access control mechanism that enforces location-based restrictions. The application relies on the authenticity of received Beacon+ advertisements to compute the relative location of an authenticated device compared to an asset and provides access to the asset data if and only if the device is within close physical proximity. In the hospital setting, nurses and physicians who are away from their personal computer but moving around with their smartphone must be within close physical proximity of a patient to access her medical record. With this scheme, an access control breach only results in a small fraction of sensitive data leak-

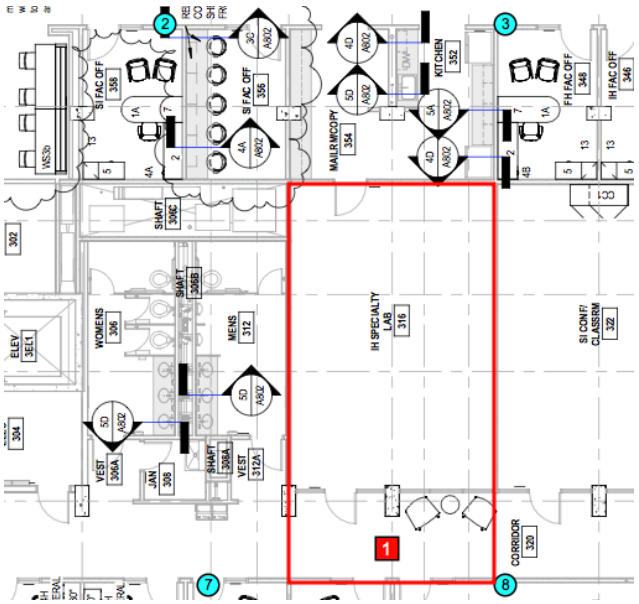

Figure 5: Example Web Application Showing Secure Real-Time Tracking System. The blue circles are Beacon+, the solid red block is a tracked device, and the red square outline is the acceptable boundary of that device.

age, since an attacker that steals an authenticated device only gets access to data that is within proximity. The location is only one factor in a multi-factor access control scheme to authenticate a user.

Implementing the location-based restrictions application requires only minor additions to the secure real-time tracking system. Personnel can use the same smartphone or BLE device they sign into for the tracking system to access sensitive data. As personnel move about the organization, the trusted server tracks their location. When the tracked device enters the close proximity of assets, the trusted server checks the credentials of the device and authenticity of the Beacon+ advertisements and sends the device the appropriate data from assets in range. Similarly, when devices leave proximity of an asset, the trusted server revokes access to that asset's data and the App removes the record[3]. The trusted server can choose the level of granularity on which to enforce location-based restrictions. For example, in the hospital context, the trusted server may choose to organize patient records based on room, rather than solely using distance as the metric. In addition, the trusted server can tailor the information sent to the devices based on the credentials of the user (e.g., physicians may be sent more sensitive information about a patient than nurses).

This approach provides location-based restrictions without the need of additional authentication at every step. While an attacker that steals one of these authenticated devices can see the sensitive information about nearby patients, the threat is not much different from the existing accepted threat in which an attacker could walk around the hospital and take the paper medical records that often sit unattended outside of patient rooms. One possibility is to have physicians re-authenticate upon entering each room which prevents an attacker from walking around with a device to get basic patient information but puts a burden on physicians and nurses. This is a trade-off between privacy and usability which can be set as desired, and the App supports both configurations.

In some cases, a physician might require accessing more details of a patient's health records or may require accessing a medical record for a patient that is not in the same room. In this case, the App on the device allows physicians to provide further forms of authentication (e.g., fingerprint, additional password) to increase their access. Note that this access is only provided temporarily each time additional authentication is provided, preventing an attacker from breaking the location-based restrictions if she steals the device. Additionally, physicians can always return to their private offices to use traditional access control techniques to gain access to a wider range of medical records.

By using location-based restrictions for access control, hospitals get the technological and convenience benefits of electronic medical records with the traditional privacy model of paper medical records, in that successful attackers only get access to localized sensitive information rather than access to a large database of many records.

Attack Mitigations. An active attacker may perform a denial-of-service attack on the tracking system to cause patient harm or thwart productivity. This attack is mitigated by having authorized individuals use additional authentication methods to bypass the location-based restrictions and temporarily gain access to patient records, or return to an authorized computer system (e.g., office computer). This type of adversary can also steal an authenticated device (i.e., a physician logged in and misplaced the device) and use it to obtain patient records via the location-based restrictions application. The application mitigates this attack by deleting patient records on a set time interval and when it moves outside the range of patients.

Note that the location-based restrictions application requires that the trusted server have knowledge of patient locations in the hospital (either at a physical location or room-level granularity). The tracking system can be made to track patient locations by associating BLE devices with patients, or the trusted server can link with existing hospital management techniques that track patient locations.

5.3 BCMA Physical Proximity Enforcement

BCMAs typically involve scanning barcodes on patients and medications to interface with electronic records. The purpose of a BCMA is to reduce medication administration errors by confirming the patient, drug, dose, route, and time. Koppel et al. [31] find that nurses and hospital staff often circumvent proper BCMA use because of malfunctioning scanners, unreadable wristbands, and emergencies. The process of circumventing BCMAs includes placing barcodes on scanners and door jams.

These circumventions put patients at risk due to the increased likelihood of incorrect barcode scanning, thus, providing the wrong medications. We conjecture that enforcing close physical proximity to a patient can mitigate this con-

[3]The App is also setup to remove data from the display after a configurable timeout, which protects against an attacker that cuts network communication in an effort to force an asset's data to persist on the screen even after moving out of range of the asset.

sequence.

Specifically, scanning devices would need to accept authenticated advertisements from nearby Beacon+s and send them to the trusted server. If the trusted server concludes that the scanner is indeed near a patient, it will return a value that enables the scanner to perform its scanning function.

Upon scanning a patient barcode, the patient name from the barcode is sent with recent advertisements to the trusted server. The trusted server will then compare the patient name it receives to the one it has mapped to that physical location. If the two match, the scan will complete. Otherwise, the scan will fail, and the operator is alerted.

6. EXPERIMENTS

We deployed eight evenly spaced Beacon+ prototypes of one side of the floor in our building to emulate a setup that would be used in typical hospital settings. Each Beacon+ was placed at its chosen location and assigned a unique ID and secret key that is shared with the trusted server. Upon startup, each Beacon+ begins broadcasting an authenticated BLE advertisement containing its unique ID, latest sequence number (monotonically increasing once per second), calibrated transmit power at 1 meter, and MAC. Advertisements are broadcast every 125µs. We experimented with several values for n, the propagation constant from equation 1, and ultimately decided on $n = 2.7$ for our experiments. It provided the most accurate measured distance from compared with the actual location of tracked devices.

We used a Google Nexus 4 smart phone as the tracked device. We created an Android App to periodically scan and collect all Beacon+ advertisements within range (aggregating the measured RSSI values for each Beacon+ ID). The collected advertisements are then bundled into a device update and sent via Wi-Fi to the trusted server, which authenticates each of the advertisements in the update and calculates the position of the device.

6.1 Tracking System Accuracy

To measure the accuracy of our Beacon+ tracking system, we placed the device at various locations and compared the calculated location from the tracking system with the actual location in the building. Initially, we measured the accuracy using the trilateration approach, using the measurements from the three Beacon+ prototypes with the strongest received signal strength for that update. However, we found that the measured signal strength from our BLE hardware contained a fair amount of noise, often causing the trilateration calculation to fail (i.e., the resulting circles created from the distance measurements did not intersect). Rather than using trilateration in our experiments, we calculated the position of devices using an approach that is less accurate, but more flexible.

Translated Midpoint Method. For each device update received, the trusted server sorts the valid Beacon+ advertisements in order of received signal strength and can calculate the device's position for this update as long as at least two advertisements are valid. If there are three or more valid advertisements, the trusted server uses the top three Beacon+ ads (based on RSSI values) and forms a triangle, with one vertex corresponding to each of the Beacon+ locations in the environment. Each vertex is then translated toward the midpoint of the opposite side of the triangle, with translation distance proportional (or in our case, equal) to the measured distance between the device and that Beacon+.

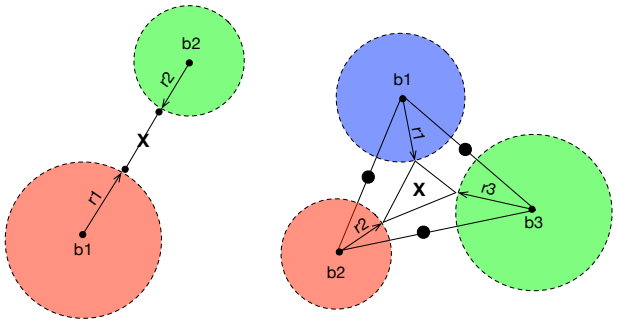

Figure 6: Translated Midpoint Method to calculate device position.

If there are only two advertisements, a line is formed between the two Beacon+ locations, and each point is translated toward the other point with a distance equal to the measured distance from the device to that Beacon+. Finally, the device's position is calculated as the centroid of the resulting triangle (in the case of three valid Beacon+ advertisements) or midpoint of the resulting line (in the case of two Beacon+ advertisements). Using the new approach resulted in position calculation with precision 1-2 meters in the best case and 9-10 meters in the worst case.

Using the translated midpoint method, the resulting Beacon+ tracking system is flexible and accurate, providing a position calculation with the precision of 1-2 meters in the best case and 9-10 meters in the worst case. Compared to the trilateration approach, the translated midpoint method achieves a better overall tracking system in the environment of our experimentation.

6.2 Power Consumption

We connected an MSP-430 LaunchPad to an Agilent programmable power supply. Since the MSP430 LaunchPad runs off of a +5V power source, we set the output voltage to 5 volts and maximum current to 1A. Our power supply showed that in the case of the MSP430 emulating an iBeacon, the power draw was between 15 and 20 mA. In the case of the MSP430 emulating a Beacon+, the power draw was between 22 and 25 mA. Therefore, the overhead of Beacon+ over a standard Beacon running on our test platform was between 20% and 46%.

6.3 Location-Based Restrictions

We created an Android App that collects and forwards Beacon+ advertisements to the trusted server and displays patient records sent in return. After validating a device and calculating its position, the trusted server compares the device position with the location of patients in the building and only sends records of nearby patients (10 meters in our experiments). When a device moves out of range of a patient, that patient record is removed from the list in the App.

For this experiment, we created a mock patient record database on the trusted server based on the OpenMRS Demo Data [26], and set the location of four of the patients in the database to locations in the building environment (yellow

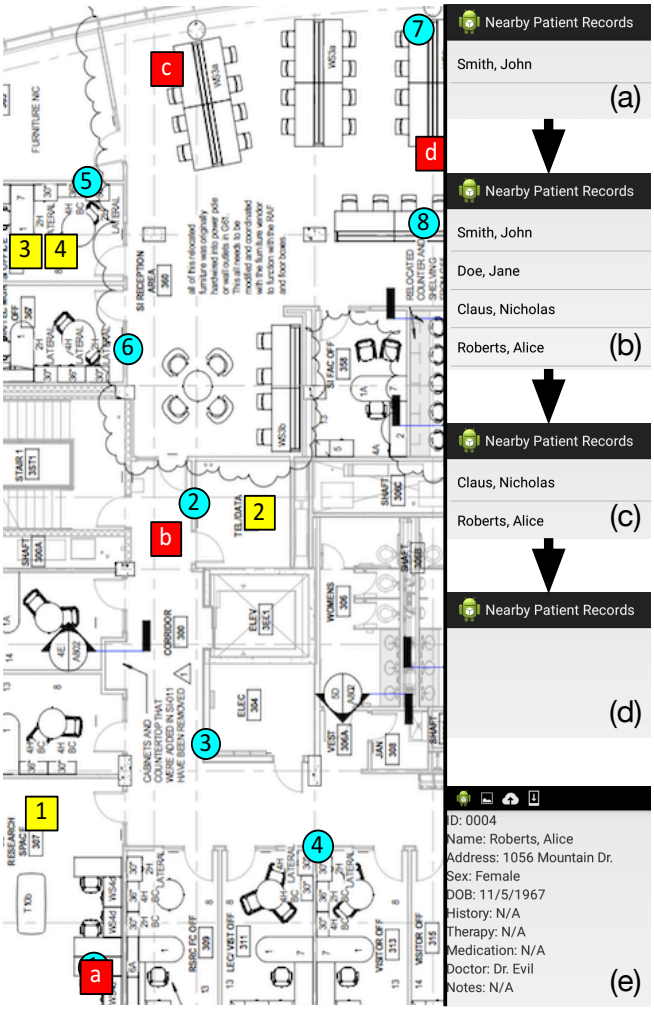

Figure 7: Location-Based Restrictions on Access Control.

squares are shown in Figure 7). Then, we walked around the building with the smartphone running the App to view the records of the nearby patients, i.e., the patients that were within 10 meters of the device's tracked position.

Figure 7 shows four snapshots (a through d) of the experiment in action. The visual GUI of the Beacon+ tracking system is shown on the right. The GUI shows the location of the Beacon+ prototypes (blue circles), the patients in the building (yellow squares), and where the device is located at each snapshot (a through d). For each snapshot in Figure 7, we also include the screen capture of the device running the patient record access App at its respective location.

7. REMOVING THE TRUSTED SERVER

We assume a central trusted authority (i.e., the trusted server) in our secure location sensing application architecture. However, this assumption is susceptible to an attacker who gains unauthorized access to the trusted server. If this should happen, all security guarantees are invalidated because the attacker would have access to the private keys of every Beacon+. Removing the trusted server is a complicated problem because Beacon+ supports unidirectional communication only; therefore, we cannot use a two-way protocol to assert trust nor can we introduce an out-of-band channel for weak authentication [14, 32].

To remove the trusted server we construct a protocol based on the timed efficient stream loss-tolerant authentication (TESLA) broadcast protocol [29]. This protocol assumes a large set of mutually untrusted receivers in a sensor network with packet loss. A sender computes the MAC t of a message with a key k known only to itself. The sender broadcasts the authenticated message m, t, and some set of receivers buffer the message. Time t later, the sender discloses the key k and the receiver authenticates the packet. This protocol is unlike our previous Beacon+ protocol because it assumes that the sender and receiver clocks can be loosely synchronized. It also introduces hash chains to authenticate keys at the receiver.

A hash chain is generated by selecting a random element s and repeatedly applying a one-way function F. We can verify any element of the chain through commitment s_i by performing $F^{j-i}(s_j) = s_i$, where $i < j$. TESLA uses hash chains to generate authentication values, k from the above, and discloses k at time t (e.g., one key per second).

We design Beacon+'s new protocol without a trusted server as follows. We initially generate a random secret s and unique ID. We then calculate $H_N = H^N(s)$ where H^N is the hash of s, N times. We put H_N and ID into a digital certificate C and sign it with a certificate authority's private key. C, s, and ID are placed on the Beacon+. At each time period i, the Beacon+ sends an advertisement containing C, ID, a message M containing the value i, a MAC on M computed with the key $H^{N-(i+1)}(s)$, and the value H^{N-i}.

The verifier in this protocol is the smartphone or medical device. The verifier collects advertisements from two adjacent time periods (i initial and j final) and checks that the advertisements are current based on its own internal clock. Next, the verifier validates C and hashes $H^{N-j}(s)$, j times, to obtain H_N. This value is in C, thus, it can be validated. The verifier then verifies time period i's MAC using the key output from time period j. We diagram this protocol in Figure 8.

We differ from TESLA in how we do synchronization. Specifically, TESLA requires a digital signature key pair on the sender and a nonce from the receiver. The receiver records the current time and sends the sender a nonce. The sender replies with its clock time and the nonce signed with its public key. Clock synchronization is useful for the receiver because it can check that the key k received has been disclosed yet. Beacon+ is strictly unidirectional, thus, cannot receive a nonce like the sender in TESLA. Instead, the verifier in our protocol can check if two adjacent time periods are current by querying the tracking server described below.

Removing the trusted server in our architecture adds new entities and roles. For example, the secure real-time asset tracking system adds a certificate authority, tracking server, and map authority (i.e., database server). We logically separate the tracking server and map authority because these components could be distributed. The certificate authority issues a signed certificate to every Beacon+. The medical device and smartphone later verify the signature on the Beacon+ certificate when receiving Beacon+ advertisements.

The tracking server allows both the medical device and smartphone to make application-specific queries to the map

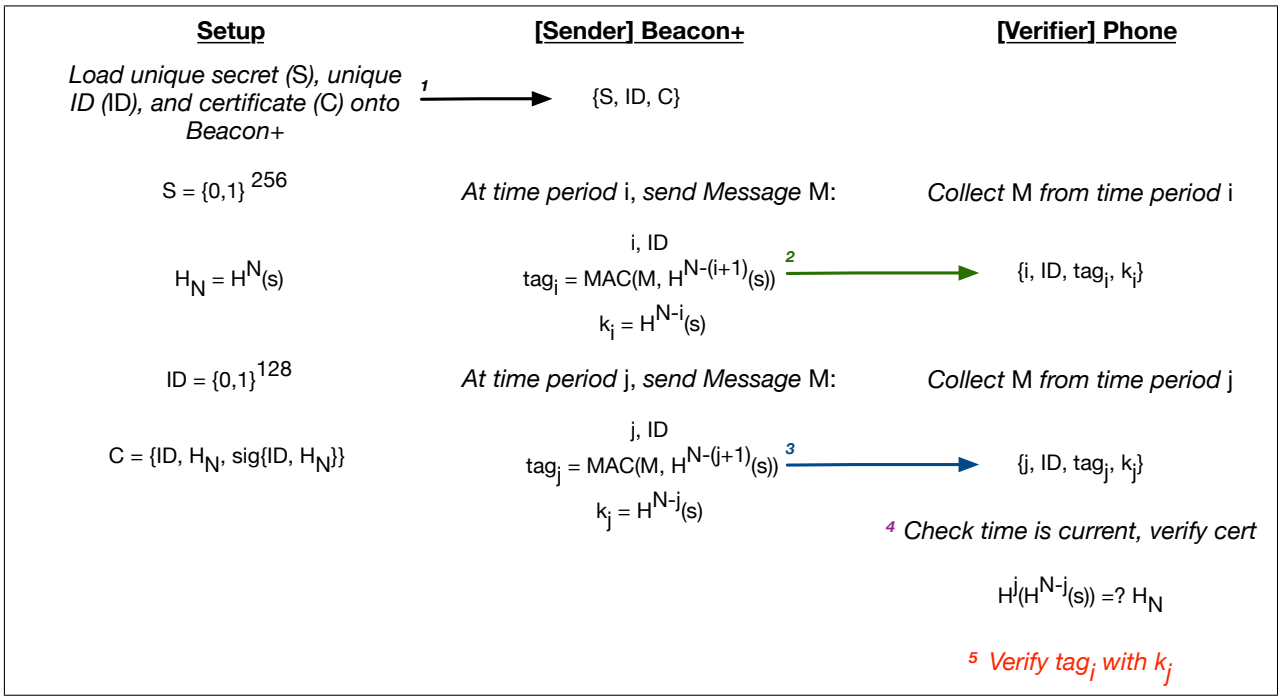

Figure 8: Beacon+ protocol without central trusted authority.

authority. For example, a medical device would send a location query [4] that contains a set of unique Beacon+ IDs, the latest time period j and k_j where $k_j = H^{N-j}(s)$. The tracking server would verify $H^j(H^{N-j}(s)) = H_N$ where H_N is in the Beacon+ certificate. If and only if verification succeeds and k_j has not been previously seen, the tracking server processes the query using the map authority and returns a result.

There exists an implicit assumption that devices that can verify Beacon+ advertisements are also trusted. We can make this an explicit assumption by requiring mutual authentication between the smartphone or medical device and the tracking server. In this case, only trusted devices can communicate with the tracking server.

8. CONCLUSION

In this work, we have shown that Beacon+ can be used to implement secure location sensing applications that have the potential to improve healthcare processes in terms of security, efficiency, and safety. We implemented a secure real-time tracking system for hospitals that also provides a foundation for a novel application that applies location-based restrictions on access control.

9. ACKNOWLEDGMENTS

This research was funded by the National Science Foundation under award number CNS-1329737. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors.

[4]We expect other queries such as location-based access queries.

10. REFERENCES

[1] E. Bertino and M. S. Kirkpatrick. Location-based access control systems for mobile users: concepts and research directions. In *SPRINGL*, pages 49–52. ACM, 2011.

[2] M. Bhuptani and S. Moradpour. *RFID field guide: deploying radio frequency identification systems*. Prentice Hall PTR, 2005.

[3] T. Bradley. Pros and cons of bringing your own device to work, Dec. 2011.

[4] R. Bruno and F. Delmastro. Design and analysis of a bluetooth-based indoor localization system. In *Personal wireless communications*, pages 711–725. Springer, 2003.

[5] Z. Chen, Q. Zhu, H. Jiang, H. Zou, Y. C. Soh, L. Xie, R. Jia, and C. Spanos. An ibeacon assisted indoor localization and tracking system.

[6] S. Contini. The factorization of rsa-140. *RSA Laboratories' Bulletin*, 10:1–2, 1999.

[7] A. Developer. Getting started with ibeacon, 2014.

[8] Ekahu. Asset tracking & management, 2015.

[9] Estimote. Estimote: Real-world context for your apps, Aug. 2015.

[10] S. Feldmann, K. Kyamakya, A. Zapater, and Z. Lue. An indoor bluetooth-based positioning system: Concept, implementation and experimental evaluation. In *International Conference on Wireless Networks*, pages 109–113, 2003.

[11] K. Finkenzeller. *RFID Handbook: Radio-frequency identification fundamentals and applications*. Wiley, 1999.

[12] Gimbal. The gimbal store, Aug. 2015.

[13] C. Gomez, J. Oller, and J. Paradells. Overview and evaluation of bluetooth low energy: An emerging

low-power wireless technology. *Sensors*, 12(9):11734–11753, 2012.

[14] T. Halevi and N. Saxena. On pairing constrained wireless devices based on secrecy of auxiliary channels: The case of acoustic eavesdropping. In *Proc. 17th ACM conference on Computer and Communications Security (CCS 2010)*, pages 97–108, 2010.

[15] Hardware Breakout. Bluetooth low energy boosterpack for the launchpad, Aug. 2015.

[16] S. Hay and R. Harle. Bluetooth tracking without discoverability. In *Location and context awareness*, pages 120–137. Springer, 2009.

[17] iBeacon for Developers. https://developer.apple.com/ibeacon/. Accessed: 2015-08-17.

[18] T. Kleinjung, K. Aoki, J. Franke, A. K. Lenstra, E. Thomé, J. W. Bos, P. Gaudry, A. Kruppa, P. L. Montgomery, D. A. Osvik, et al. Factorization of a 768-bit rsa modulus. In *Advances in Cryptology–CRYPTO 2010*, pages 333–350. Springer, 2010.

[19] M. Kouhne and J. Sieck. Location-based services with ibeacon technology. In *Artificial Intelligence, Modelling and Simulation (AIMS), 2014 2nd International Conference on*, pages 315–321. IEEE, 2014.

[20] T. Labs. Trusted beacon reference, Apr. 2015.

[21] I. LiveViewGPS. Gps tracking - tracking systems - you can trust, 2015.

[22] D. Manolakis. Efficient solution and performance analysis of 3-d position estimation by trilateration. *Aerospace and Electronic Systems, IEEE Transactions on*, 32(4):1239–1248, Oct 1996.

[23] P. Misra and P. Enge. *Global Positioning System: Signals, Measurements and Performance Second Edition*. Lincoln, MA: Ganga-Jamuna Press, 2006.

[24] W. Murphy and W. Hereman. Determination of a position in three dimensions using trilateration and approximate distances. *Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado, MCS-95*, 7:19, 1995.

[25] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil. Landmarc: indoor location sensing using active rfid. *Wireless networks*, 10(6):701–710, 2004.

[26] OpenMRS Wiki Rosources - Demo Data. https://wiki.openmrs.org/display/RES/Demo+Data. Accessed: 2015-08-17.

[27] S. K. Opoku. An indoor tracking system based on bluetooth technology. *arXiv preprint arXiv:1209.3053*, 2012.

[28] S. A. Ortiz and L. M. Ortiz. Systems and methods for tracking assets using associated portable electronic device in the form of beacons, Mar. 2014. US Patent App. 14/194,953.

[29] A. Perrig, R. Canetti, J. D. Tygar, and D. Song. The tesla broadcast authentication protocol, 2002.

[30] M. Portnoi and C. Shen. Location-aware sign-on and key exchange using attribute-based encryption and bluetooth beacons. In *IEEE Conference on Communications and Network Security*, pages 405–406, 2013.

[31] K. R, W. T, T. JL, and K. BT. Workarounds to barcode medication administration systems: their occurrences, causes, and threats to patient safety. *J Am Med Inform Assoc*, 15:408–423, 2008.

[32] K. B. Rasmussen, C. Castelluccia, T. S. Heydt-Benjamin, and S. Čapkun. Proximity-based access control for implantable medical devices. In *Proc. 16th ACM conference on Computer and Communications Security (CCS 2009)*, pages 410–419, 2009.

[33] N. Sastry, U. Shankar, and D. Wagner. Secure verification of location claims. In *Proceedings of the 2Nd ACM Workshop on Wireless Security*, WiSe '03, pages 1–10. ACM, 2003.

[34] F. Schrooyen, I. Baert, S. Truijen, L. Pieters, T. Denis, K. Williame, and M. Weyn. Real time location system over wifi in a healthcare environment. *Journal on Information Technology in Healthcare*, 4(6):401–416, 2006.

[35] B. Specification. Version 1.1. *Includes: IMS Learning Resource Meta-data Information Model IMS Learning Resource Meta-data XML Binding Specification IMS Learning Resource Meta-data Best Practice and Implementation Guide Available at: www. imsproject. org*, 2001.

[36] Texas Instruments. MSP430FR5969 launchpad development kit, July 2015.

[37] F. Thomas and L. Ros. Revisiting trilateration for robot localization. *Robotics, IEEE Transactions on*, 21(1):93–101, Feb 2005.

[38] R. Want. Near field communication. *IEEE Pervasive Computing*, (3):4–7, 2011.

[39] J. Yang, Z. Wang, and X. Zhang. An ibeacon-based indoor positioning systems for hospitals. 2015.