For Your Eyes Only

Robin Kravets
University of Illinois at
Urbana-Champaign
rhk@illinois.edu

Güliz Seray Tuncay
University of Illinois at
Urbana-Champaign
tuncay2@illinois.edu

Hari Sundaram
University of Illinois at
Urbana-Champaign
hs1@illinois.edu

ABSTRACT

As users interact with an Internet of Things (IoT) ecosystem, they leave behind traces of information about their presence, preferences and behavior. While the ecosystem can track individuals' movements to provide enhanced recommendations, individuals have little control over how this information is being used or distributed. Such tracking has led to increasing privacy concerns over the use of IoT. While it is possible to develop systems to enable anonymous interaction with IoT, anonymity results in limited benefits to both individuals and IoT ecosystems. In response, we present Incognito, a secure and privacy preserving IoT framework where user information exposure is driven by the concept of identity. In particular, we advocate user-managed identities, leaving the control of the choice of identity in a given context, as well as the level of exposure, in the hands of the user. Using Incognito, users can create identities that work only within certain contexts and are meaningless outside of these contexts. Furthermore, Incognito allows for simple management of information exposure through contextual-policies for sharing as well as querying of an IoT ecosystem. By giving individuals full control over the information traces that they leave behind in an IoT infrastructure, Incognito, in essence, puts individuals on equal footing with the entities that want to track their behavioral data. Incognito fosters a symbiotic relationship; users will need to expose information in exchange for personalized recommendations and IoT organizations who provide sophisticated user experiences will see enhanced user engagement.

Categories and Subject Descriptors

C.2.1 [**COMPUTER-COMMUNICATION NETWORKS**]: Network Architecture and Design–*Wireless communication*

Keywords

Bluetooth Low Energy, Identity, Internet of Things, Privacy

1. INTRODUCTION

Every day, users are interacting with hundreds and thousands of devices in both intentional and unintentional ways. Currently, these devices are being linked through local and cloud services to form an Internet of Things (IoT). As users interact with this IoT in stores, museums and other public spaces to find useful localized information, they leave breadcrumbs in the form of information traces about their presence, preferences and behavior. By intentionally exposing pieces of their personal information, users could benefit from complex services and enhanced interactions. Additionally, organizations, including retail locations and museums, can provide sophisticated benefits in exchange for this information. However, to prevent unintentional leaks of personal information, users must be able to manage their information exposure. To this end, the users and organizations need to collaborate through an IoT ecosystem that benefits both the users and organizations, while allowing the users to protect their personal information.

To achieve the full potential of such an IoT ecosystem, the breadcrumbs collected in an environment must be associated with a user. As more and more information is collected about a user, more and more refined recommendations can be made. By tracking individuals and their data, the information in their individual data traces can be aggregated into meaningful business intelligence, allowing users, companies and organizations to leverage the vast potential of IoT. The benefits of exposing user information and interacting locally with the physical entities in a user's environment, as well as the IoT ecosystem as a whole, can be immense. These benefits can range from simple scenarios that improve a user's tour of a museum (*e.g.*, "What did other people with an interest in impressionism see here?") or shopping experience in a grocery store (*e.g.*, "What do other people who are on a diet buy here?"), to more complex scenarios that can help a user navigate through a foreign city (*e.g.*, "What did my parents order when they visited this café?"; "What do locals like to do?"). Additionally, by allowing users to query the IoT ecosystem, they can look back at their own traces to see what they have done in the past. To benefit from any of these examples, users must be willing to expose some amount of information about themselves to help provide personalized recommendations.

However, as new technologies are deployed, users are still unsure of how to interact with the world around them, or if they even want to. Even though users are afraid of exposing too much personal information, it is clear that they are willing to expose some information if they receive concrete benefits (*e.g.*, EZpass provides faster and discounted road toll payments, frequent buyer supermarket apps give fuel discounts and free food). Although we are already seeing many new applications in this direction, there is an all or nothing approach to information exposure. Instead of exposing their identity all of the time, when shopping, a user may only want to expose that they are vegetarian to help them navigate through a store. By exposing a little more information about their identity and shopping history, the user may be given new suggestions for what to buy or even access

MCS'15, September 11, 2015, Paris, France.
© 2015 ACM. ISBN 978-1-4503-3545-4/15/09 ...$15.00.
http://dx.doi.org/10.1145/2802130.2802137.

to special sales. However, a user may not want to expose all of their personal information in a given context. They may even want to go so far as to interact anonymously.

Contemporary culture, with dystopian visions of a future where individuals seamlessly interact with the physical world around them (*e.g.*, the film "Minority Report") mirrors individual's fears of information exposure. In a typical scenario, powerful corporations and government agencies track individuals as they interact with their environment. Although these visions of the future are very off-putting to many users, they are not so far from the current capabilities of data aggregators. Indeed these capabilities, including the user's security concerns have started to raise concerns about IoT with policy makers [1, 2].

These concerns about unauthorized information exposure are in part related to the current web-advertising framework where advertising networks use third-party cookies to track an individual's web-browsing [3]. At any given time, there are several hundred entities tracking any individual's behavior over the web. This behavioral information is aggregated and traded over real-time exchanges. Additionally, these data aggregates can be used for unexpected purposes other than for advertising, for example, to deny someone credit [4]. While many websites let visitors know that they use third-party cookies, individuals have very little idea about the extent of information that is gathered about them by third-party tracking entities and are unable to query entities about what is stored about them. Finally, user queries to determine what information has been gathered about them and sharing of such information is similarly infeasible. In the context of IoT, if the control of a user's data is entirely left to external organizations, users will remain skeptical and likely forgo the use of any IoT ecosystem, limiting the benefits to all parties involved.

In response, we envision a radically different future where individuals are in full control of their information exposure, including traces that they leave behind with any part of the IoT ecosystem. In particular, in exchange for intentional exposure of limited information, individuals can access unique, complex services, beyond product recommendations or advertisements. However, currently, there is no simple way for a user to manage their exposure and how the exposed information is re-used.

In this paper, we present the design of Incognito, a framework where user information exposure is driven by the concept of identity. When interacting with the IoT ecosystem using Incognito, individuals, not the ecosystem, are in control of the "identity" they expose. By giving individuals full control over the information traces that they leave behind in an IoT infrastructure, Incognito, in essence, puts individuals on equal footing with the entities that want to track their behavioral data. While identity is currently being addressed in IoT systems, the main focus is the identity of the things [5], not of the users.

To enable flexible management of user information, Incognito allows each user to generate multiple identities or pseudonyms, based on their context—location, domains, personal state, and time—that we term *contextual identity*. (`cid`)[1]. For example, the user could have one identity for each store they visit, or a new identity for each time they visit a store. The user can then limit information exposure and aggregation by managing access to their breadcrumbs on a per-identity basis. Additionally, if they want to disconnect from one of their identities, they simply stop using it and create a new identity, thus enabling a limited form of "digital forgetting."

Essentially, each user controls how much information is passed on, and so potentially stored and used for recommendations, by the ecosystem by setting location- and app-specific identities inside Incognito. Incognito manages all communication with the IoT ecosystem, eliminating information leaks to the apps running on a user's mobile device. Furthermore, with Incognito, an individual's data stored in the IoT ecosystem is available to that person via an authenticated query, as well as to friends to whom the individual has granted access. Given the increasing concern over the use by advertisers of third-party cookies to track individuals' web-browsing [3], we believe that putting control of a user's information exposure in the hands of the user is critical to widespread adoption of IoT by the public.

2. BUILDING AN IOT ECOSYSTEM

The success of an IoT ecosystem depends on the services it provides and the interactions it has with the users. To understand the challenges associated with designing and deploying an IoT ecosystem, we assume that a user moves through environments carrying a smartphone that interacts with the IoT ecosystem, via Wi-Fi, and periodically advertises their presence, via Bluetooth Low Energy (BLE). In return for exposing some level of personal data, the ecosystem provides the user with location- and context-specific information that will enhance the user's experience. While we discuss the core system design challenges and design principles in this section, it is important to remember that any solution must be simple for the user. Complex systems are difficult for users to understand and manage and they might frequently interrupt the user so much that they decide to stop using the services the environment provides.

2.1 Context Discovery

To bootstrap the interactions between the user and the ecosystem, the user, or more specifically their device, needs to be able to determine what environment or context they are currently in. For example, consider our user Alex, who walks into a café. To discover the associated context beyond GPS- and map-based correlations, previous work has suggested using ambience features such as light and audio as well as acceleration to characterize user's movement in that context [6]. In SurroundSense [7], the authors implement ambience fingerprinting (*e.g.*,fingerprinting light, audio, acceleration, Wi-Fi etc.) to achieve logical localization (*i.e.*, determining context). Context discovery can also be achieved without performing ambience fingerprinting, where crowdsourcing is used for the extraction of semantic location from sensor data (*e.g.*, audio, image), without any need for fingerprinting [8]. However, this type of approach relies on heavy computer vision techniques (*e.g.*, scene classification, optical-character-recognition, object recognition), as well as heavy natural language processing (*e.g.*, speech recognition, sound classification). Although context discovery is a key component to enabling context-based IoT successful, this is not the focus of this paper. For the rest of this paper, we assume that there exists a context/environment discovery mechanism and the IoT app on Alex's smartphone can determine that she just entered a new environment, the CoffeeShop. Additionally, we assume that the user has access to a service that maps the location to an authoritative server for that location/environment.

2.2 User Identity

As users interact with an IoT ecosystem, they must present the ecosystem with an identity, which the ecosystem uses to track the user's movement and behavior. Obviously, not every user wants to expose the same amount of information about themselves, if they want to expose anything at all. Although many identity management systems have been proposed, *we advocate user-managed identities, leaving the control of the choice of identity, as well as the level of*

[1] In the rest of the paper, we shall use the word "identity" to refer to contextual identity and we shall use `cid` to make explicit this connection.

exposure, in the hands of the user. In this context, the challenge lies in the effective choice of an identity in a given environment and the level of personal information that is exposed through the use of this identity.

The interaction between a user and an environment must be symbiotic. By tracking a user, the environment can learn patterns and behaviors to enhance its ability to perform (*i.e.*, sell more goods, attract more users). Similarly, there needs to be benefit to the user (*i.e.*, discounts, better recommendations) from exposing their personal information. Given that an IoT ecosystem can track any user that enters an environment, the choice of an identity allows the user to control their exposure in response to their expected benefits. In our café example, Alex can choose to visit the CoffeeShop without making use of the IoT services provided, visit anonymously and receive free Wi-Fi access or expose some amount of personal information in exchange for a free biscotti. In our vision, the ultimate goal of an IoT ecosystem is to allow every user to choose their identity in order to control how much information they want to expose at any given time in any given location based on their knowledge of the benefits provided by the environment.

A user can manage their information exposure by using a pool of pseudonyms, each of which can be used as an identity in a given context. The use of pseudonyms has been studied in the context of WLANs [9] as well as MANETs [10], and more specifically in vehicular networks [11]. However, in these domains, the pseudonyms are chosen to strictly provide privacy and are never reused. By allowing the reuse of pseudonyms in IoT ecosystems, a user can intentionally expand their exposure and use the same identity in multiple environments. For example, as a frequent patron of the CoffeeShop, Alex would like to maintain her IoT history across all of her visits. She can do this by using the same pseudonym for every visit. Her friend Brian, on the other hand, would only like the CoffeeShop to be able to track his session today and not connect it to any other visits. Brian can maintain his level of exposure by using a new pseudonym for every visit.

2.3 User Identity Use and Reuse

As a user interacts with the IoT ecosystem, their communication and devices leave traces of their interactions. If this communication is not managed correctly, even the use of pseudonyms cannot hide the user's real identity. On the ecosystem level, when a user decides to interact with the IoT ecosystem, the user must register their presence for this visit (or session) with the environments's IoT cloud services. After registration, the user can interact with IoT beacons placed around the environment. IoT beacons can be simple devices that advertise some location-based information or more complex devices that interact directly with the IoT app on the user's smartphone.

Simple communication between the user's smartphone and the IoT environment has the potential to expose the user's identity, and so other personal information. At the lowest level, user identity information can be tied to a user's device identity (*e.g.*, MAC address). Essentially, user's interaction with the IoT devices can be traced back to the user simply by tracking their device. Similarly, other credentials transmitted during communication can be spoofed and replayed by malicious users/nodes [12]. By aggregating all information at a higher level, a large amount of information about a user can be tracked. For example, a system administrator can easily associate IP addresses to MAC address in WLANs [13], tying a user's interaction with Internet services to their device and identity. This can also be achieved in the context of IoT, by associating access to beacon-advertised websites (*e.g.*, by UriBeacons [14]) to the user's MAC address. In other words, whenever a user clicks on a link that was advertised by a specific beacon, it creates an opportunity for the server to associate the user's MAC address to the IP address from which the website was accessed. Even IEEE 802.11 has many implicit identifiers in the "supposedly" secure messages [15]. Even though the design of an ecosystem may look correct on the outside, it is very important to understand the underlying technology that is being used and the effect on exposure.

Although an IoT ecosystem needs to be designed to enable the user to limit how much information is exposed, it is also important to allow the user to expose more information if they believe there is more benefit to them for doing so. Essentially, the user should be able to control the scope at which their information can be accessed and aggregated. In the context of IoT, there are two interesting scopes: exposure across multiple locations of the same organization and exposure across multiple organizations. Within the same organization, the user should be able to control the level of sharing. For example, the next day, Alex visits a different CoffeeShop location. For this visit, she wants to have the option of keeping her session traces local to this CoffeShop location or linking it to her visits to all CoffeeShop locations. Again, it should be up to the user to manage the scope of exposure of their data. A user should also be able to control cross-environment exposure. Consider that Alex visits CafeAuLait. Although not the same organization as the CoffeeShop, Alex would like to link all of her IoT sessions for all visits to cafés. This collaboration can also enable new business agreements between different organizations [16].

2.4 Sharing/Querying

Since the ecosystem is collecting all of this information about a user, it could be beneficial to the user to be able to access their own data. Additionally, to support social networking-based recommendation systems, users should be able to share their own data without exposing any information about who they are sharing it with. In our café example, Alex's friend Brian is interested in what food and drink Alex liked at the CoffeeShop. Alex would like to share her IoT traces with Brian in a secure way. A user should be able to share their data if they decide they trust the other user. However, it is beyond the scope of our work to deal with situations where the friend becomes malicious and no longer trustworthy.

2.5 Secure Validation and Privacy

Finally, all of the communication between the user and the IoT ecosystem must stand up to some level of security and provide privacy for the user. The specific requirements will depend on the design of the ecosystem. However, given that the user is broadcasting their presence, we believe one of the most important security components for IoT lies in validating the user. Consider Chris, another customer at the CoffeeShop, who tries to impersonate Alex and add incorrect information into the CoffeeShop's IoT ecosystem. This may be possible if Chris can steal Alex's identity information during broadcasts and replay this information. The design of the ecosystem must ensure that a user cannot be unknowingly replicated. Elliptic curve cryptography (ECC) based approaches have been proposed for identity authentication in RFID systems [17], as well as for the more general Internet of Things [12, 18]. However, it might not be ideal to run these complex cryptographic functions on resource-constrained IoT devices.

3. IOT ECOSYSTEM ARCHITECTURE

Our main goal is to enable users to manage their experience with the IoT ecosystem by controlling the amount of information they expose to the ecosystem in each environment. In this context, we propose Incognito, a privacy-preserving IoT ecosystem architecture that allows a user to share any desired part of their identity within a

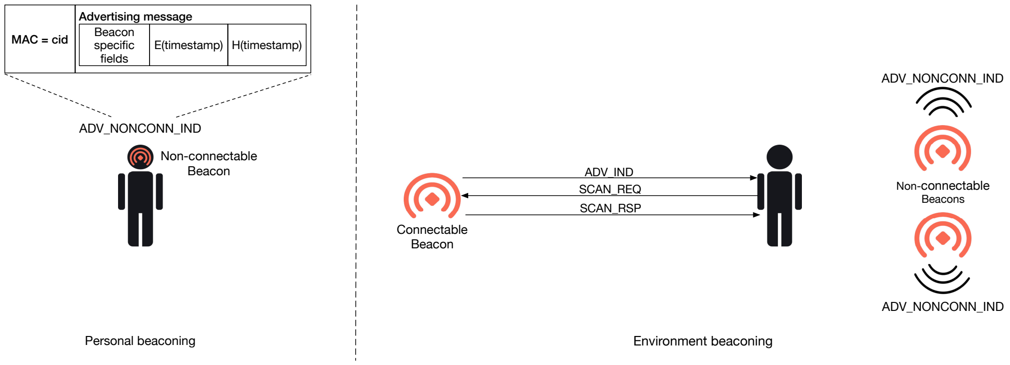

Figure 1: Interacting with BLE-based IoT infrastructure. BLE Beacons advertise with ADV_IND (connectable) or ADV_NONCONN_IND (non-connectable) broadcast messages. Receivers of ADV_IND messages respond to the sender with SCAN_REQ, and beacon responds with SCAN_RSP to the sender. Users can choose to switch to non-connectable beaconing (*i.e.*, personal beaconing) to show their presence.

specific environment. Incognito allows uses to select specific identities in a given context and provides privacy-preserving mechanisms for interacting with the ecosystem.

3.1 User and Infrastructure Devices

With Incognito, there are devices that belong to each environment and devices that belong to the users. A user carries a smartphone, enabled with Wi-Fi, Bluetooth Low Energy (BLE) and other data services. The Wi-Fi or cellular data is used to register with the authoritative server in the environment.

As the user moves through the environment, they listen to beacons broadcasting information about the environment. These devices can range from small tags emitting simple beacons via BLE that users can scan to more complex servers that the users interact with via Wi-Fi or LTE in the environment's cloud. For this paper, we focus on traditional WiFi communication with the ecosystem and BLE beacons broadcasting in one of two modes, passive scan or active scan (see Figure 1). BLE beaconing is used by both beacons in the environment to advertise localized information and by users to advertise their presence.

Environment beacons can use either mode. In passive scanning mode, the BLE advertising node periodically sends a beacon (or advertising message ADV_NONCONN_IND), which can contain up to 31B of user data. An example of this type of passive beacon is the UriBeacon, which broadcasts a URI that the user can follow to get more information. Although 31B may be sufficient for some applications, many applications may need to send more data. To support this, advertising nodes can use active scanning mode and send active beacon messages (ADV_IND). Any device receiving a active beacon responds immediately with a scan request message (SCAN_REQ) and the advertising node finishes the data exchange with a scan response message (SCAN_RSP), again with a maximum of 31B of application data. Given the information in the beacon messages, the user can contact the environment's server for more information.

We expect users to advertise their presence using passive beacons. In this case, the 31B beacon is sufficient. The environment listens for these user beacons and collects traces about the movement of the user.

3.2 Environments

An environment in our ecosystem encompasses the space and devices used by an organization (*e.g.*, store, museum, park). Although an organization may maintain multiple spaces (*e.g.*, multiple stores of the same company, or multiple parks in a city), these environments can share user IoT information within their own cloud. For example, Company X in Figure 2 has two stores that share the same cloud. However, company X and company Y cannot share information about a user if they use different identities in the different stores (see discussion of *cid* below).

Since the number of users in an environment will change dynamically, along with their locations and interests, the environment should be able to adapt the information within its beacons. Given the limited amount of space in any beacon message, the dynamic information should be simple, most likely using only a few bytes. For example, the message could include statistics of interaction with that item, as well as a short category id. The statistics of the interaction as well as the category id could let the users know whether they should follow the information (*e.g.*, the URI) of the beacon. The presence of this limited information could potentially lead to more efficient apps on the user's smartphone (*i.e.*, not having to look up the URI if the category is not of interest).

3.3 Users

The users in our ecosystem move through different environments, advertising their presence as needed via a BLE beacon and interacting with the devices in the current environment. Similar to the environment, the user can also advertise meta-data to the environment based on context. For example, an individual who has just come into the grocery store after a yoga session, can include the keywords `{yoga, tired}` in their message and the environment could then respond to the beacons by showing statistics of items bought by other individuals sharing the same context. The environment could also put items not of interest to this individual in low-power mode if there is no other nearby foot traffic.

To limit their exposure in any given environment, each user can manage the "identity" that they expose at a given time and place, interacting with the environment using a context-based identifier (`cid`). Each user maintains their list of existing `cid`s and preferences in different environments on their device or in their own personal cloud. The lifetime and reuse of a `cid` determines how much information a given environment can track about each user, as well as limit the ability of different environments collaborating to share one user's data. This use of a `cid` leaves the management of information exposure in the hands of each user via the following options:

- `Anonymous`: Random `cid` every x sec: The environment can only track a user for x sec, after which the user appears as someone new.

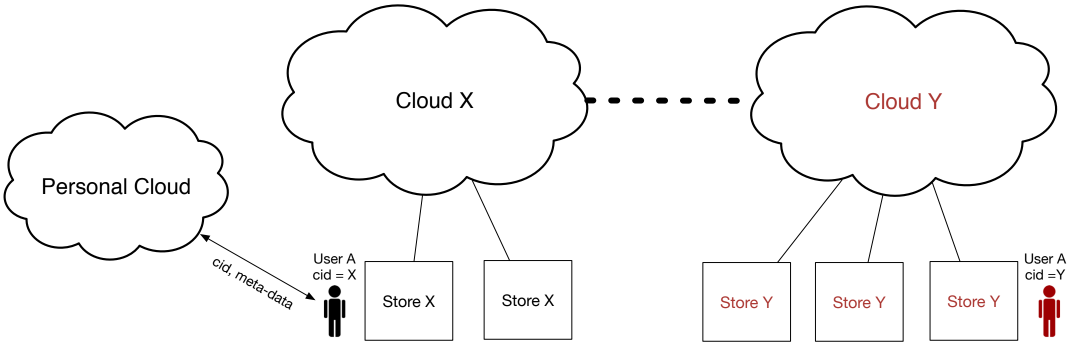

Figure 2: Privacy preserving IoT ecosystem. A single user can use different pseudonyms for different contexts. If the two stores belong to the same company, the user can create a single pseudonym to use in both stores. If both companies belong to the same category, the user can set their `cid` for X and Y to be the same, creating the information link indicated by the dashed link.

- `Local-One-Time`: Random `cid` per session: The environment can track the user for the whole time they are present. However, they cannot find a connection between multiple sessions from the same user in the same environment over time.

- `Local`: Random `cid` per environment: The environment can track a user's visits over time, but cannot find a connection to the same user in a different environment.

- `Cross-Domain`: Random `cid` per environment class: Multiple environments of the same type can track and share user information within that environment class.

- `Global`: Global `cid`: A user exposes their main identity all of the time.

The simple use of BLE to broadcast the user's `cid` in an environment, as well as interact with some of the BLE beacons, would ultimately leak the user's presence everywhere in the world. To prevent this, BLE allows a device to rotate through random MAC addresses at varying timescales. Although our approach works with these random MAC addresses, the payload of a BLE beacon message is quite small and the `cid` could take up a significant part of it.

Instead, Incognito sets the BLE MAC address to the `cid`, eliminating the need for the `cid` to be in the beacon message body. Additionally, if the user's device interacts with a BLE device in the environment, the messages will contain the user's device's MAC address. If a random MAC address is used, the user's presence is hidden from the environment and so would not allow the environment to track the user. Instead, if the MAC address is set to the user's `cid`, the environment can track who is interacting with their devices.

It is important to note that a user can be tracked by their Wi-Fi MAC address as well. Although BLE and Wi-Fi share the same frequency space, there is no overlap in functionality. Therefore, Incognito sets Wi-Fi and BLE devices to use the same MAC address to provide identity support across both technologies. Although there is no physical reason why both BLE and Wi-Fi can't use the same MAC address, it will likely require some modifications to the user's devices.

3.4 User-Managed Information Sharing

There are three levels of information sharing in our ecosystem: user-environment; user-user; and environment-environment, as can be seen in Figure 3.

3.4.1 *User-to-Environment Sharing*

To enable user-environment information sharing, a user exposes their `cid` to an organization, which can then track the user. Since the user is managing the identities, it is up to the user to provide the capability of authenticating their `cid`. A user could use a (`PublicKey`, `PrivateKey`) pair for all interactions. However, the use of the same `PrivateKey` across all `cids` would allow the ecosystem to link all of the `cids` from the same user. Instead, for each $\texttt{cid}_i$, the user creates a public-private key pair: ($\texttt{PublicKey}_i$, $\texttt{PrivateKey}_i$).

Each time a user enters a new environment, the user communicates over a secure channel to register with the organization (see Figure 3). Registration includes the following tuple: ($\texttt{cid}_i$, $\texttt{PublicKey}_i$, $\texttt{PrivateKeyEncrypt}_i(\texttt{cid}_i)$). As the user moves around the environment, it interacts with the devices in the environment.

To advertise their presence, the user broadcasts their `cid` over BLE. If the `cid` is broadcast by itself, a malicious user could listen to a user's broadcast `cid` and replay it anywhere, essentially impersonating the user. To prevent this replay attack, the broadcast includes the following: ($\texttt{cid}_i$, `Hash(TimeStamp)`, $\texttt{PrivateKeyEncrypt}_i(\texttt{TimeStamp})$). Since only the user can encrypt the timestamp with their private key, other users cannot impersonate them. Additionally, by using an increasing timestamp, a broadcast can only be used once. The hash of the timestamp is only used to reduce the amount of data needed in the broadcast message.

Once the environment decrypts the `TimeStamp`, it can compare the hash of the decrypted `TimeStamp` with the hash in the message. We show the updated content of user's advertising messages in Figure 1. Although a shared key could be created for a `cid`, the use of a public/private key pair allows the user to reuse the `cid`, and so the associated public key, in multiple contexts without exposing the private key.

3.4.2 *User-to-User Sharing*

To enable user-user sharing and allow trusted friends access to their data, a user can share their $\texttt{cid}_i$ for a specific environment. The friend can then use the shared `cid` to query the environment without exposing their own identity. To achieve this, the initial user communicates the following over a secure channel: ($\texttt{cid}_i$, $\texttt{PrivateKeyEncrypt}_i(\texttt{cid}_i)$). The friend includes this information in the query. The environment can validate the query by decrypting with $\texttt{PublicKey}_i$ and comparing the sent $\texttt{cid}_i$ to the stored $\texttt{cid}_i$. This simple interaction prevents random users from querying the environment's database.

To include limited access for the friend, the initial user would send the following over the secure channel: ($\texttt{cid}_i$, $\texttt{PrivateKeyEncrypt}_i(\texttt{cid}_i$, access_id, access_param)). The access_id can be used to control access for a particular transaction based on the access_param, which, for example, could be once, for the next x seconds or unlimited. For this to work, the environment would need to keep all access_ids forever. Notice that using the same protocol, an individual can query the IoT infrastructure to obtain the data that the infrastructure has stored about themselves.

3.4.3 Environment-to-Environment Sharing

To allow the user to control exposure from environment-environment sharing, they can select the appropriate cid in the given environments. By using different cids, the environments cannot connect the user across environments, as can be seen in Figure 2. However, if the user decides to use a environment-class-based cid, the user is allowing all environments in that class to share their information.

4. IMPLEMENTATION CHALLENGES

Current mobile platforms (e.g., Android, iOS) and the BLE protocol itself pose challenges that need to be handled in order to implement Incognito and our proposed IoT ecosystem architecture.

There are currently a limited number of devices that are equipped with the full functionality of BLE. Essentially, only a few smart devices can operate in broadcast mode (i.e., capable of beaconing). Additionally, it is not possible to change the MAC address on BLE-enabled smart devices, even if the devices are rooted. Mobile platforms do not expose the necessary APIs at the moment to achieve this even though it is part of the Bluetooth LE protocol.

Fortunately, vendors have started to include full BLE support in their devices due to the increasing popularity of Internet of Things. We expect an increase in the number of fully compliant BLE devices with the proliferation of IoT products and applications. One open question is how to manage switching between different BLE modes (*e.g.*, from broadcast to client) and how it will affect the performance of BLE discovery.

In our current prototype, we use Nexus 5 smart phones (that currently do not support broadcast and peripheral modes). For beacons, we use the nRF51822 Bluetooth Smart Beacon Kit [19] produced by Nordic Semiconductor. By default, these beacons are in accordance with Apple's iBeacon format [20]. The Nordic SDK supports all the possible BLE modes for their beacons, as well as other devices Nordic BLE devices. Additionally, Nordic devices support changing the MAC addresses on the beacons manually or periodically to any valid MAC address. Since not all devices can beacon and listen at the same time, to represent a single user, we pair a Nordic beacon with a BLE-enabled smart device, creating the illusion that user's device can operate in any BLE mode and that we can change user's MAC address for broadcasts.

5. USE & IMPACT

For Incognito to be successful, it must be easily integrated into a user's smartphone system and apps. If these interactions are not designed correctly, a user's personal information could be exposed to a company's app on the user's phone and so eventually the company itself. However, it is import to understand the impact of these design decisions on the potential benefits to the the users and the organizations. Therefore in this section, we discuss how Incognito integrates into a user's system, systems settings and IoT apps. Then, we discuss the possible impact of Incognito on end users, retail organizations and advertisers.

5.1 Incognito on Smartphones

To support IoT apps and manage user information exposure, the user installs the Incognito system component on their smartphone. The main role of Incognito is to manage the user's cids, the associated keys and the environments they are used in. Incognito internally manages the shared meta-data and the access control rules for IoT apps that make use of Incognito and also exposes an IoT widget to help each individual set policies for interacting with the IoT ecosystem.

In Incognito, cid management is triggered by changes to context, typically changes in location. Run-time cid management is based on the user's choice of exposure level. For example, if the user sets the mode of interaction with the IoT infrastructure as "Local-One-Time", any application requesting a cid receives a new cid, and the individual uses a new public key, private key pair for this session.

The flexibility of Incognito allows a user to set the number of visits to a location for which Incognito would use a one time identity before the user decides to use a local or domain identity. Essentially, if the policy says that the user must visit the same physical location two times before using a local cid, for the first two visits, Incognito would select one-time cids that anonymize the individual's visit. If the visits two different retail locations belong to the same chain, say a grocery store chain, the user could also have a policy that allows IoT apps to receive the same cid. To facilitate sharing with other users, a policy can be created that allows friends to query the IoT infrastructure about this user's interaction traces.

Over time, as individuals visit a large number of physical spaces, Incognito must manage many identities. The key usability challenge is not in managing identities but in making it easy to set policies that govern information exposure. We note in passing that today, popular smartphone OS's allow users to set very simple polices that govern the exposure of location information; we expect to see rapid progress in the design of mobile interfaces to set and manage polices.

5.2 IoT Apps

The final pieces of the ecosystem are the smartphone apps that interact with the environment through Incognito. Without any control, the IoT app could save all of the cids used for that particular app and so bypass any control the user intended for exposure. Essentially, the IoT app could track the user over multiple sessions by simply storing all of the cids used by this user and subsequently transmitting them to the server. To prevent such unintentional exposure, Incognito does not expose the cids to the apps. Instead, all communication with the IoT ecosystem is performed by Incognito, including communication with the IoT server in the environment and any BLE beacons. The app is thus a "front-end" to the data that it receives from Incognito.

When the IoT app is opened—triggered by a change in location or context—it provides an IP address to Incognito and asks Incognito to register itself with the environment's server. At this point, Incognito chooses an appropriate identity consistent with its policy setting. Incognito then sends the tuple {cid, PrivateKeyEncrypt(cid), PublicKey} to the server over WiFi or any other data channel. This exchange registers the IoT app with the environment's server using the cid specified for this session. As noted at the end of Section 3.3, Incognito ensures that the BLE and WiFi devices share the same MAC address to prevent users from being tracked by the organization.

To enable the user to specify more information about themselves, each app provides each user the ability to set a few keywords appropriate to the app. For example, a user could set the following keywords as part of their identity within the grocery application: {Male, Vegan, Millennial}. Again, these keywords are communicated to the environment's server by Incognito. The key point here is that these keywords are dynamically set and the user could potentially change the keywords every time they visits the store,

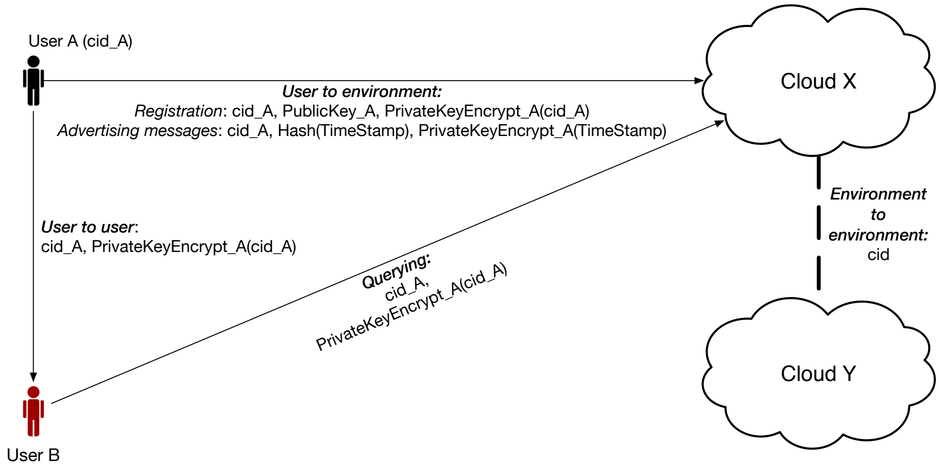
User A (cid_A)
User to environment:
Registration: cid_A, PublicKey_A, PrivateKeyEncrypt_A(cid_A)
Advertising messages: cid_A, Hash(TimeStamp), PrivateKeyEncrypt_A(TimeStamp)
Cloud X
Environment to environment: cid
User to user: cid_A, PrivateKeyEncrypt_A(cid_A)
Querying: cid_A, PrivateKeyEncrypt_A(cid_A)
Cloud Y
User B

Figure 3: Sharing pseudonyms with the IoT ecosystem. *User to environment*: User registers cid, encrypted cid and public key with the cloud. *User to user*: User can share cid and encrypted cid with other trusted users. *Environment to environment*: Different environments can share cids if a cid is created for a category of environments. Also, *User to environment querying*: Other users can query the cloud with cid and encrypted cid to retrieve their friend's experiences.

thus altering their persona. If the user maintains the same cid, the organization will develop a more detailed view of this person, since it will aggregate information across visits.

To explain this further, consider how two IoT apps— one for a grocery store (e.g. WholeFoods, an U.S. grocery chain) and one for travel (say the official tourist application from Paris)—on the user's smartphone interact with the policies set in Incognito. In the grocery store scenario, besides having the familiar functionality to search, see product recommendations and browse for products, IoT apps supported by Incognito will be able to provide richer, interactive shopping experiences. In particular, IoT apps will enable more natural text query scenarios: "What is popular with women?"; "What did people who are vegan buy?" This combination of IoT apps and Incognito will also be able to provide social recommendations—products or services from people similar to the user, either in terms of the keywords exposed by these people or in terms of their behavioral similarity (the aisles they browse, the specific categories of products or services in which they show interest). In the tourist scenario, we can also envision a rich, immersive experience with an IoT ecosystem. For example, besides providing recommendations, a user could ask questions such as: "Where does a person from the U.S. with a $50 dinner budget eat?"; "Where do people with my tastes in fashion shop?" If one is willing to accept that the infrastructure stores data for long periods of time, one could ask when at the Louvre: "What did my parents spend time watching at the Louvre?", assuming that the person asking the question has obtained the appropriate {cid, PrivateKeyEncrypt(cid)} pair for their parents. In all of these scenarios, the app passes the query, or the request for recommendations, to Incognito to communicate with the server and subsequently receive the data, perhaps in the form of XML, to render in the application.

It may be tempting to simply use familiar data mining algorithms [21, 22] or community discovery algorithms such as [23] at the infrastructure to answer these questions. However, resolving these questions within our dynamic identity framework is challenging. First, individuals can easily change their cid, de-linking current behavioral data from prior data. The temporal variation in the privacy settings of people frequenting a location influences the quality of the recommendations. Second, depending upon the policy settings in Incognito, it is possible that the behavioral data available is specific to this location only, precluding the possibility of aggregating behavioral data across the IoT ecosystem. While cids reduce the amount of data to be analyzed at a location, this reduction comes at the cost of weaker, less specific recommendations.

5.3 Impact

Incognito has impact on end users as well as organizations that maintain IoT infrastructures. Our main claim about Incognito is that it levels the playing field between users who seek sophisticated retail and physical space experiences and organizations that track "digital foot-traffic" to provide services and recommendations to their customers. It does so through user-managed identities, leaving the control of the choice of identity, as well as the level of exposure, in the hands of the user. Importantly, from a system's standpoint, since Incognito works well within the BLE specification, it should be straightforward to implement as soon as more BLE devices implement the full BLE protocol and subsequently expose it via an api.

From an individual's perspective, Incognito brings with it a sense of control over their data. The most significant impact is on one's sense of privacy. Not only can they interact anonymously with any given IoT infrastructure, but also, in case an organization running the IoT infrastructure with whom they have used a persistent identity is not providing useful experiences, they can easily change their identity. Incognito also makes it simple to manage a user's identity through context-triggered policies. Furthermore, a user can determine what any IoT infrastructure has gathered about them as well as behavioral information, as well as share their information with trusted friends. Finally, there is a sense of security through the use of well-understood and highly scrutinized public key infrastructures.

From an IoT organization's perspective, while at first blush there appears to be a loss of "control," there are three important benefits to using Incognito. First, for a brand that provides high-quality retail experiences, their customers will trust the brand and be willing to share more information about themselves. Thus, such brands will not only develop better marketing strategies, but also develop better products for their customers. Second, allowing individuals to control their information exposure levels the playing field amongst different retail organizations. Rather than investing in a costly "information race" with other retail organizations to know as much as possible about the customer through proprietary tracking technologies, organizations will only compete based on the quality of the IoT experience that they provide to the customers. We expect Incognito to enable a robust competition between brands—IoT organizations (*e.g.* different grocery retail chains) will need to compete for customers based on the sophistication of their recommendations as well the overall retail experience based on their customers' IoT interactions. It is possible that when organizations provide poor experiences, customers will choose to interact anonymously, thus providing less value to such IoT organizations. Finally, since individuals, not IoT organizations, are responsible for identity, there is diminished responsibility for the individual's privacy since this person can trivially disconnect from their past digital traces.

6. FUTURE DIRECTIONS

Given the explosion of IoT applications and environments, Incognito gives individuals complete control over how much of their identity is exposed in a given context. Incognito assures that a user's identity cannot be re-used by malicious users even if it gets stolen, by using a simple digital signature technique. This technique can be integrated into IoT infrastructures in a seamless way, and it takes very little space in the IoT communications. Additionally, Incognito gives individuals the opportunity to query the infrastructure for their own traces or for the traces of the members of their close social network to learn about their past behavior in a secure way.

Although the crucial components in Incognito can be implemented today, the full system requires a clean design and implementation. Hence, our first goal is to fully-implement Incognito using off-the-shelf smartdevices and IoT products (e.g., BLE beacons). In order to bring the IoT ecosystem into reality, a context-awareness component is of absolute necessity since it can enable automatic identity switching based on context without requiring user intervention after registration. Furthermore, we are planning to work on a "contextual identity"-based recommendation system and quantify the trade-off between recommendation quality and identity exposure.

7. ACKNOWLEDGMENTS

This research is funded in part by a Focused Research Grant from Google, Inc.

8. REFERENCES

[1] Natasha Singer. F.T.C. says internet-connected devices pose big risks. The New York Times, Jan 2015. http://nyti.ms/1MPWgby.

[2] The Economist. The internet of things (to be hacked), July 2014. http://econ.st/1FQ3A1Y.

[3] Joseph Turow. *The daily you: How the new advertising industry is defining your identity and your worth*. Yale University Press, 2012.

[4] Lori Andrews. Facebook is using you. The New York Times, Feb 2012. http://nyti.ms/1GGOG46.

[5] David Meyer. Samsung invests in internet of things identity management platform evrythng, Oct. 2014. http://bit.ly/1JoTivU.

[6] Hans Gellersen, Michael Beigl, and Albrecht Schmidt. Sensor-based context-awareness for situated computing. In *Proc. of Workshop on Software Engineering for Wearable and Pervasive Computing.*, 2000.

[7] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. Surroundsense: mobile phone localization via ambience fingerprinting. In *Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 261–272. ACM, 2009.

[8] Yohan Chon, Nicholas D Lane, Fan Li, Hojung Cha, and Feng Zhao. Automatically characterizing places with opportunistic crowdsensing using smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 481–490. ACM, 2012.

[9] Ming Lei, Xiaoyan Hong, and Susan V Vrbsky. Protecting location privacy with dynamic mac address exchanging in wireless networks. In *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*, pages 49–53. IEEE, 2007.

[10] Julien Freudiger, Mohammad Hossein Manshaei, Jean-Yves Le Boudec, and Jean-Pierre Hubaux. On the age of pseudonyms in mobile ad hoc networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.

[11] Rongxing Lu, Xiaodong Li, Tom H Luan, Xiaohui Liang, and Xuemin Shen. Pseudonym changing at social spots: An effective strategy for location privacy in vanets. *Vehicular Technology, IEEE Transactions on*, 61(1):86–96, 2012.

[12] Parikshit N Mahalle, Bayu Anggorojati, Neeli R Prasad, and Ramjee Prasad. Identity authentication and capability based access control (iacac) for the internet of things. *Journal of Cyber Security and Mobility*, 1(4):309–348, 2013.

[13] Qi He, Dapeng Wu, and Pradeep Khosla. The quest for personal control over mobile location privacy. *Communications Magazine, IEEE*, 42(5):130–136, 2004.

[14] Uribeacon. http://uribeacon.io.

[15] Jeffrey Pang, Ben Greenstein, Ramakrishna Gummadi, Srinivasan Seshan, and David Wetherall. 802.11 user fingerprinting. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, pages 99–110. ACM, 2007.

[16] Konstantinos Lampropoulos and Spyros Denazis. Identity management directions in future internet. *Communications Magazine, IEEE*, 49(12):74–83, 2011.

[17] Sheikh Iqbal Ahamed, Farzana Rahman, and Endadul Hoque. Erap: Ecc based rfid authentication protocol. In *Future Trends of Distributed Computing Systems, 2008. FTDCS'08. 12th IEEE International Workshop on*, pages 219–225. IEEE, 2008.

[18] Guanglei Zhao, Xianping Si, Jingcheng Wang, Xiao Long, and Ting Hu. A novel mutual authentication scheme for internet of things. In *Modelling, Identification and Control (ICMIC), Proceedings of 2011 International Conference on*, pages 563–566. IEEE, 2011.

[19] nrf51822 bluetooth smart beacon kit. http://bit.ly/1L34QCz.

[20] Apple. Getting started with ibeacon, 2014. http://apple.co/1MPb7CU.

[21] Albert Bifet, Geoff Holmes, Bernhard Pfahringer, and Ricard Gavaldà. Mining frequent closed graphs on evolving data streams. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 591–599. ACM, 2011.

[22] Hong Cheng, Xifeng Yan, and Jiawei Han. Mining graph patterns. In *Managing and Mining Graph Data*, pages 365–392. Springer, 2010.

[23] Yu-Ru Lin, Jimeng Sun, Hari Sundaram, Aisling Kelliher, Paul Castro, and Ravi Konuru. Community discovery via metagraph factorization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(3):17, 2011.