Hindawi
Mobile Information Systems
Volume 2017, Article ID 4725858, 11 pages
https://doi.org/10.1155/2017/4725858

Research Article

Enhanced Internet Mobility and Privacy Using Public Cloud

Ping Zhang,[1] Mimoza Durresi,[2] and Arjan Durresi[1]

[1]*Department of Computer and Information Science, Indiana University-Purdue University Indianapolis, Indianapolis, IN, USA*
[2]*European University of Tirana, Tirana, Albania*

Correspondence should be addressed to Arjan Durresi; durresi@cs.iupui.edu

Received 1 December 2016; Revised 17 April 2017; Accepted 2 May 2017; Published 13 June 2017

Academic Editor: Laurence T. Yang

Internet mobile users are concerned more and more about their privacy nowadays as both researches and real world incidents show that leaking of communication and location privacy can lead to serious consequence, and many research works have been done to anonymize individual user from aggregated location data. However, just the communication itself between the mobile users and their peers or website could collect considerable privacy of the mobile users, such as location history, to other parties. In this paper, we investigated the potential privacy risk of mobile Internet users and proposed a scalable system built on top of public cloud services that can hide mobile user's network location and traffic from communication peers. This system creates a dynamic distributed proxy network for each mobile user to minimize performance overhead and operation cost.

1. Introduction

Mobility support has been one hot topic for the last decades, and many designs are proposed aiming to implement an "anywhere, anytime" Internet connectivity experience. Although research community provided a number of designs, such as [1–3], today's majority Internet mobility support is done through cellular service providers: mobile device receives a private IP address that is routable within cellular service provider network, and Internet traffic will go through a nearby cellular Internet Gateway to public Internet. The IP address issued by cellular network is allowed to roam across limited distance and time; until then a new IP will be assigned and may also be accompanied with changing of Gateway. Existing connections have to be terminated then reinitiated by the mobile devices. Due to the nature of intermittent connection and nonpublic routable address behind the gateway, it is difficult to resolve mobile device's network location and initiate a connection to it by peer host itself. To address this issue, major mobile OS vendors and application vendors implemented "push notification" to emulate an on-demand message pushing service, such as Apple Push Notification Service (APNS) [4], Google Cloud Messaging (GCM) [5], or Microsoft Push Notification Service (MPNS) [6]. Under the hood, mobile devices keep live connections with Push service providers to receive real-time message. When one mobile user wants to communicate with another user, the message is delivered through either Push service, or leverage Push service to bootstrap a direct connection between mobile device and peer node.

Even though these existing infrastructures enabled various mobile applications today, it does not solve problems including privacy vulnerability and lack of general reachability. When a mobile node connects to its peers, connections are set up on its exposed public IP address, which is either its actual public Internet attach point or a gateway close to its physical location. That means from its exposed IP address all of its peers can identify the approximate geolocation of the mobile node. Even worse, peers not only can track the trajectory of the mobile node from its IP changes but also can capture with high fidelity the movement timings to precisely model the location history and movement pattern of the user using this mobile node. Those pieces of information can be further used to project its future location statistically [7–9]. On the one hand, this type of prediction can be useful for certain purpose [10], but for a concerned privacy user, it is not good news.

To protect privacy, existing research works had focused on increasing anonymity of collected user location data [11], limiting shared location information, or evaluating privacy

exposure level before sharing location data [12]. However how to protect privacy for mobile node's direct communication did not draw equal attention. Unfortunately, any website can track their user's IP history and run all kinds of analysis and data mining to model user's behavior. Mobile applications step one level further that can accurately track a single user's movement and can form a precise network address timeline, even when application is not granted access to GPS location. Additionally, any mobile app providing direct communication exposes mobile node's location history not only to the mobile app vendor but also possibly to all other contacts using the same app.

The general availability is another issue of cellular-based Internet mobility. Besides not being available to devices that are not cellular network equipped, Push notification systems are centralized proprietary services that different systems are not compatible with others. For example, to send a message through APNS, both sender and receiver must be able to talk to APNS and have APNS client installed. Also, the Push sender must register with APNS priorly. A device with only MPNS and another device only having APNS will not be able to leverage Push service to communicate. To enable cross Push system communication applications have to manage the identity mapping and communication channel translation themselves with extra external services. Push notification as an indirect communication mode, nevertheless, cannot solve privacy issue solely. Due to the architecture limitation, it can only be used to send a small piece of data, that is, 4 KB as current standard. If peers want to use high-bandwidth communication such as video stream, a direct connection not through Push service must be created separately. Additionally Push services are usually OS/vendor bounded and without any legacy support. Existing applications cannot benefit from Push service unless reconstructed. Usually, it is not an easy task and expensive as communication model is different.

VPN has become a popular service as more and more Internet users start to concern about their privacy. Through either private VPN service or multiple relay networks like TOR, Internet users can hide where they are when they communicate with peer hosts or websites and also obscure who they talk to from their ISP. Limitations are that today's VPN services do not provide particular support of mobility and add performance overhead as traffic always goes through a static relay end host. The overhead will increase when the mobile host moves to different network while still using the same relay point. Additionally VPN services usually are not designed to handle incoming connection well.

To solve these privacy and connectivity problems we proposed a system that combines mobility support and proxy and strategically creates a dynamic proxy network for a mobile node to achieve the best balance between privacy, performance, and cost. A centralized SDN controller manages relay servers in multiple public cloud data centers, and proxies are dynamically allocated on demand to form a proxy network for each mobile node. All connections between the mobile node and peer nodes are through proxies that are close to peer nodes, and as a result both the real network location and mobility characteristic are hidden completely from peer nodes. Additionally mobile node can enjoy mobility support on any Internet applications.

In the following sections, we discuss attack models of privacy attacks in Section 2. Then in Section 3 we describe the system design. Then we show results of preliminary simulation in Section 4 and list related works in Section 5. At last, we conclude in Section 6.

2. Privacy Attack Models

We assume Alice, the attack target, carrying a mobile device with her all the time so that the network/geolocation of her mobile node is approximate of Alice's geolocation. The adversary Bob wants to know Alice's current geolocation and location history so he can take advantage of that. The more accurate location history Bob knows about Alice, the more sophisticated attack he will be able to craft. In the following sections, we list four major distinct attack models that Bob can leverage to attack Alice's privacy. Note that different attack models could be combined in certain circumstances to further enhance attack effects, as described in attack scenarios.

2.1. Attack Model 1: Direct Connection Attack. By accomplishing this type of attack, Bob can successfully directly connect to Alice's device and even maintain a connection to it. Attack succeeds when connection can be set up successfully so that Bob can acquire the current location of Alice. For protocols only bound to a network location, such as TCP, an adversary might need to perform further communication to confirm Alice's device identity. Identity bound protocols, such HIP, may give Bob enough information for identify verification with just connection attempt. There is one precondition of this attack that Bob has to know Alice's network location before connecting.

2.2. Attack Model 2: Location Registry Attack. By accomplishing this type of attack, Bob can indirectly acquire Alice's network location from a registration service, such as DNS, without direct interaction with mobile node. A successful attack will reveal one temporary contact point (not necessarily real network location of mobile node such as when Proxy is leveraged) and give Bob chance of further verifying by attempting direct connection. One precondition of this attack is that Bob has to know Alice's network identity priorly.

2.3. Attack Model 3: Historical Location Attack. By accomplishing this type of attack Bob can collect a sequential list of where Alice has been, which can be used to profile Alice or aid other types of attacks. The sequence here is important as the more precise the location sequence is, the better the resolution of profiling adversary can be achieved. However a location list with completely wrong sequence may still be useful to Bob to some extent. The preconditions of this attack are that (1) Bob has to know Alice's network identity priorly and (2) Bob can retrieve a subset list of Alice's location history.

2.4. Attack Model 4: Location Change Timing Attack. By accomplishing this type of attack, Bob knows Alice's device handover time, that is, when Alice moves from one location to next location. This attack by itself does not reveal privacy that much, but when it is combined with other attack models, Bob can dramatically increase profiling precision and can multiply the privacy attack damage.

2.5. Attack Scenario 1: Adversary Directly Connects to Mobile Node. The simplest yet most impactful attack on Alice's privacy is that Bob can keep a live connection directly to Alice's mobile node device. Therefore, Bob will be able to know exactly Alice's network attachment location which can be mapped to geolocation. Also Bob will know when Alice's address changes. Having that Bob not only knows the real-time location of Alice but also can create a history timeline of Alice's movement. This is a combination of Attack Models 1, 3, and 4.

When Bob knows Alice's real world identity, with Alice's real-time location and historical location information he can launch all kinds of sophisticated attack or even threatening Alice's physical world safety. Without knowing Alice's real world identity Bob can still easily profile Alice by knowing her unique location history. Note that Bob does not need to be a friend of Alice to be able to trace her. Bob can be a website Alice is used to visiting, or just a script embedded in an advertisement.

2.6. Attack Scenario 2: Adversary Resolves Mobile Node's Address via a Location Service. Based on Scenario 1, assume Alice enhances her security by deploying a local firewall on her mobile node to refuse connection from Bob. This would to some extent prevent Bob from acquiring real-time location information of Alice. However, there could be some public location service, such as DNS, that can be used to resolve Alice's identity to her location for Alice to be connected. Bob can keep sending location resolution requests to this service to collect Alice's location history. This is a combination of Attack Models 2 and 3.

Compared to Attack Scenario 1 Bob's tracking capability is limited: first Bob will not be able to get deterministic real-time location of Alice since he cannot directly connect to her; second since the location registration always lags behind and is sometimes protected by throttling mechanism, Bob will not perceive precise timing or even complete location history of Alice. In this case it only accomplishes Attack Model 2. When the registration service has access control and Bob is not whitelisted to resolve Alice's address, he will not be able to track Alice. However, maintaining a whitelist is difficult and expensive, as modern Internet host usually has tens or hundreds of open connections to web servers and other hosts at any moment. On the other hand, if Bob is allowed to connect to Alice or allowed to resolve Alice's location, Alice's exposure is no different than Attack Scenario 1.

2.7. Attack Scenario 3: Adversary Connects through Proxy Moving along with Mobile Node. Alice can protect her location privacy while keeping connectivity by sending/receiving traffic through a proxy. In this case, Bob can communicate with Alice at any time, but only the proxy location is exposed to Bob. Bob will only observe proxy's location history, and, under most circumstances, Bob will not be able to detect whether Alice is behind a proxy. This is a combination of Attack Models 3 and 4.

A typical example is cellular data network. When Alice uses the cellular network to access the Internet, usually Alice's mobile host will be assigned a private network address that is routable within carrier's network, and Alice will have to route her traffic through her cellular carrier's Internet gateway for Internet access. For Bob, he will only see Internet gateway's network address as Alice's exposed network address. In this case, the carrier's Internet gateway becomes a de facto proxy. When Alice roams away new private network address will be assigned. When this new private address is associated with another Internet gateway which is usually close to the cell Alice is in, Bob will observe connection interruption and location change.

3. A Mobility Support System Protecting Privacy

Our system is based on our previous design of Mobility Support System (MSS) [13]. In this section we will briefly describe system design and discuss how it can protect privacy while providing mobility support efficiently.

3.1. Parties and Entities. There are three different parties in the system: mobile node, peer node, and Mobility Service Provider (MSP). *Mobile node* is the mobile device hosting mobile user's identity and applications. It roams across different networks and continuously communicates with its peers. During its movement, mobile node keeps changing its network attach point and exposes different public network address (such as public IP address) at times. While mobile node receives new network addresses, its network location may shift small distance (e.g., handover across a cell) or shift relative large distance (such as switching to a different service provider or performing a vertical handover). Most of the time the geolocation of the old and new address is relatively close, though the radius can be from street blocks to adjacent cities. All mobile nodes will have MSS client deployed, which handles all incoming and outgoing traffic. Connections between mobile node and proxies are identity based, and all traffics are tunneled through these connections. MSS client daemon book keeps all connections made through it to other peers and its connections to MSP control plane and also maintains a table for all proxies it connects to.

Peer nodes are Internet host on the other end of the connection. They can be a website, an ordinary host, or another mobile node. They can be categorized as either MSS deployed host that is mobility aware and connects based on identity to proxy or legacy host that only connects to IP address and locates mobile node by DNS name.

MSP manages a fleet of servers, called *Virtual Routers*, that are dynamically allocated and released from public cloud service provider's data centers. Each Virtual Router can host multiple *proxies* that relay different mobile nodes' traffic, up to one Virtual Router's resource limit. One proxy only serves

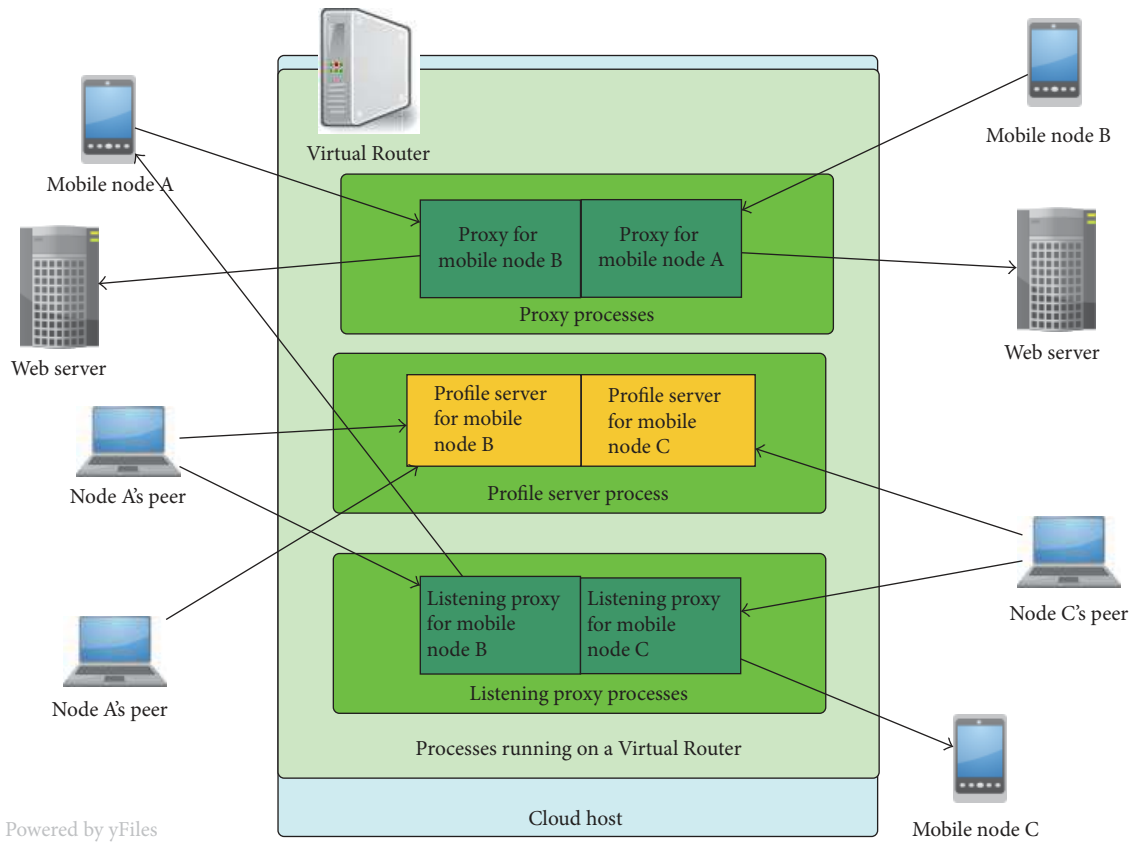

Figure 1: Virtual Router hosting proxies.

one mobile node at a time, though it can handle multiple traffic streams (or tunnels) for same mobile node. Proxy also functions as profile server that serves mobile node's profile or DNS name lookups. For each mobile node, its proxies together create an overlay network to propagate control signals. Figure 1 shows a Virtual Router hosting multiple proxies.

At any moment mobile node has a master proxy which is created or designated at a location close to mobile node. This proxy could be used for relaying traffic with low privacy concern connections or close by peer nodes, but its major tasks are to manage the proxy overlay network and delegate communication between mobile node and SDN controller. When mobile node moves away from master proxy, a new master proxy will be created to take over the task.

3.2. Proxy Allocation and Deallocation. The proxy servers handle both outgoing and incoming connections. The outgoing connection is initialized by mobile node sending request to SDN controller through master proxy. SDN controller will then select the best candidate from mobile node's current proxy fleet that has minimum communication cost to peer node or create new proxy limited by current resource availability and customer's SLA. The Virtual Router, which is assigned to host this proxy, then just chooses a random outgoing port and uses its own address to create connections to peer node. Then it instructs the selected proxy to update/initialize and take over the connection while waiting for tunnel opening request from mobile node. Incoming connection means proxy must listen on a given port for incoming connection requests. Therefore, incoming connections are exclusive, especially for legacy connection since one specific listening port can be exposed for only one mobile node on a Virtual Router. Listing proxy servers must be created priorly and are dynamically adjusted according to recent address queries, amortized management algorithm, and also the historical statistic. Furthermore, because the resource is scarce, listing proxy is more expensive than outgoing proxy and popular ports (such as 80) are more expensive than nonpopular ones. When creating connections to peer nodes, proxy can create mobility aware connection or legacy connection depending on peer node's type. In the case of mobility aware connection proxy can migrate live connections to another proxy, while for the legacy connection proxy must keep serving it until it closes.

When all connections of a proxy are closed, it becomes a candidate for removal. The decision is made by master proxy, given the current load and topology of all mobile node's proxy, user's SLA, and dampening algorithm. Once a proxy is removed, master proxy reports to SDN controller and the resource on corresponding Virtual Router is freed up for other customers. Similarly, MSP manages its Virtual Router fleet at a larger scope with the same strategy. It removes unoccupied Virtual Router or creates new ones to

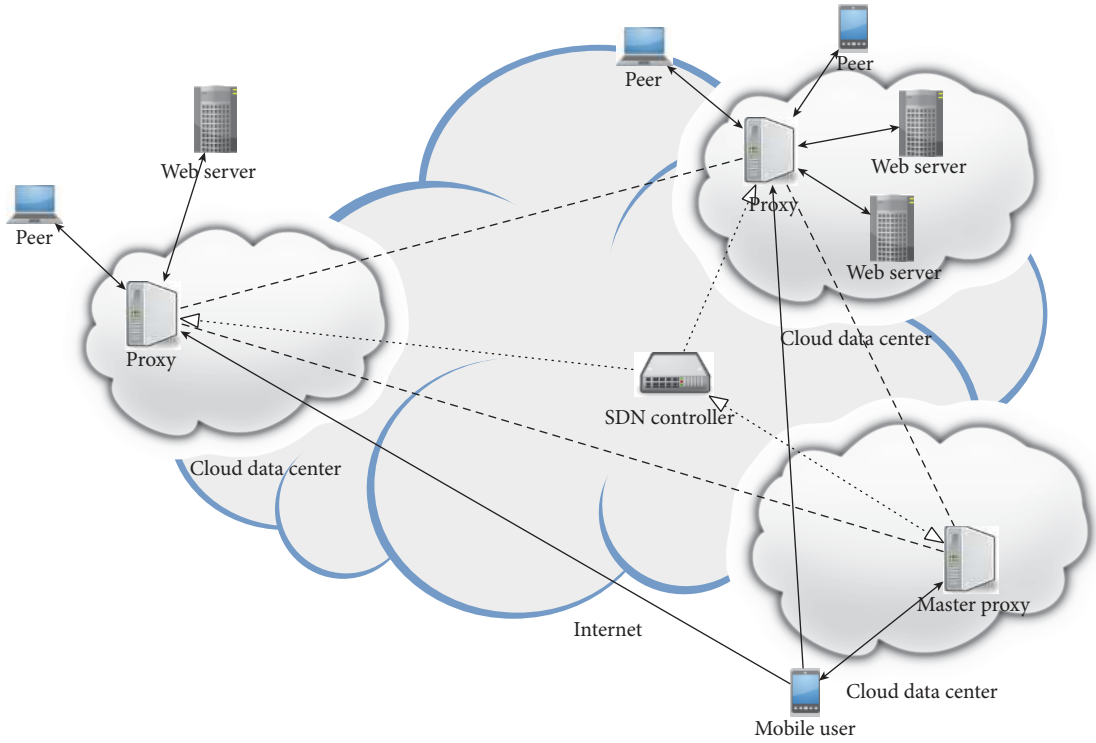

Figure 2: Proxies forming overlay network.

maintain a healthy load ratio and global presence. Public cloud enables this architecture that Virtual Routers can be created/removed in almost all the main areas around the world, in the manner of on-demand that Virtual Router can be allocated or removed within minutes dynamically.

3.3. Proxy Overlay Network. For a mobile user, MSP creates a proxy overlay network on top of all involved Virtual Routers. This proxy network only consists of proxy servers for this particular mobile user. Only master proxy exchanges control signal with mobile node. Proxy servers communicate among themselves to aggregate metrics into master proxy. Overlay network topology is self-adapted, and adding/removing proxy is requested by master proxy and controlled by SDN-style controller.

All proxy servers connected through overlay network actively sync copies of mobile node's profile and location information, even if when they do not have live connection with mobile node (as "standby proxy"). Therefore, every proxy server can serve lookup and connection request for mobile node independently. Periodically, proxy servers exchange traffic, connection, and lookup statistics. Furthermore, such statistics are summarized at a master proxy and then reported to the MSP controller.

Each mobile node has its own proxy overlay network. Different proxy networks would share the same Virtual Routers as long as there is no resource contention (e.g., a host can only listen on TCP port 80 for legacy clients for a single mobile user). However, different proxy networks do not communicate nor know others existence as their signaling and data plane are completely isolated from each other. On the other hand, Virtual Router knows all proxies deployed on it and their corresponding proxy network IDs. MSP controller has a global picture of every proxy overlay network and every Virtual Routers.

Figure 2 shows an example scenario: MSP's SDN controller has allocated three proxies for a single mobile user, and all communication to mobile host is through these proxies. The three proxies form a virtual overlay network that replicates mobile user's profile and aggregates its performance metrics. Meanwhile, proxies are exposed as mobile user's identity so the mobile user can accept incoming connections at any time from any other Internet endpoints, no matter legacy or not.

Master proxy needs to periodically review the overlay network to keep a good balance or running cost, performance, and availability. It also needs to prepare to compensate for potential future traffic. Once it determines the overlay network topology needs to change (without active action from mobile node) such as adding, deprecating, or removing proxy it will report to SDN controller.

3.4. Multihoming. Since at any moment a mobile node can be behind a few proxies, by nature MSS mobile node is considered multihoming. Its peers generally will only see one exposed network location of the mobile node, but it is also possible that a peer node connects a mobile node through two or more different proxies, especially when these connections are set up long time apart.

3.5. Both Behind Proxies. A special scenario is that both end hosts are mobile nodes behind proxies. Additionally they may belong to different MSPs which additionally limits the data to optimize performance. When two mobile nodes belong to the same MSP, since MSP controller knows locations of both, MSP will choose one "pivot" point between them to optimize for performance. If one mobile node knows the other end is also a mobile node, it may leverage that to detect how far away the other mobile node is away from it. To mitigate that, MSP controller must set a lower bound of route path length, to avoid choosing a pivot point too close to a mobile node. When mobile nodes belong to different MSPs, both only exposed proxy to the other side, and the traffic will go through two proxies. It could result in an inefficient trombone routing unless MSPs can cooperate to share some location knowledge.

3.6. Metrics. There are three major metric types that MSS system optimizes for: *privacy*, *cost*, and *performance*. In the following equations, we use m, n, and p to represent mobile node, peer node, and proxy server; use M, N, and P to represent the corresponding set.

Privacy. To quantify network location privacy protection in end-to-end communication scenario, we propose two metrics: *distance* and *timing*. After all, privacy can be quantified as the uncertainty of the mobile node's location observed from peer side. So in the case when peer believes that the exposed location is the real location, the uncertainty is 100%. However, since we cannot quantify how much peer node believes the mobile node's exposed location, we will always assume that peer node knows mobile node is behind a proxy, and the proxy is on a strategic point that will not introduce unreasonable latency penalty, that is, at some point along the route between mobile node and peer node. (Our metrics can also quantify privacy and performance of artificially away proxy as well; i.e., proxy is selected far away from mobile node and peer node, to create an illusion of being away for peer node, increasing the performance penalty.)

Distance, λ, is measured by the distance of exposed network location and the actual network location of mobile node. Distance stands for two different types of measurement in MSS: network distance and geographical distance. Network distance can be measured by network hops of end-to-end connection, or hops of network segments such as Autonomous System (AS). Geographical distance is measured by the distance of corresponding geographic locations of exposed network address and actual network address. This metric bears similarity to distance error described in [14, 15]. For example, a mobile node in New York City with IP address 128.59.*a*.*b* talks with a peer node in Los Angeles with IP address 128.97.*x*.*y*, via a proxy server in Indiana with IP address 129.79.*m*.*n*. The network distance is the hop distance between 128.59.*a*.*b* and 129.79.*m*.*n*, and the geographic distance is about 700 miles between New York City and Indianapolis. Distance measures how far away the exposed network location is from mobile host's real network location. It is a derived metric rather than simply counting number of hops because more hops do not necessarily mean larger network distance. Since adversary can derive network organizations and geographic location from network address, we define distance as how different the exposed network address is in terms of relative geographical distance, which is derived from the mapping of network address to registered geographical locations.

In general the larger the distance, the better the privacy. We use function $\mathrm{dis}(x, y)$ to represent the approximate location between network attach points x and y. Then for a given combination of mobile node (m), peer node (n), and proxy server (p), location privacy $\lambda m, n, p$ can be evaluated as follows:

$$\lambda m, n, p = \frac{\mathrm{dis}\,(m, p)}{\mathrm{dis}\,(m, n)}. \tag{1}$$

On the other hand, the performance overhead ϕ is quantified as follows:

$$\begin{aligned} \phi m, n, p = {} & \mathrm{distance}\,(m, p) + \mathrm{distance}\,(p, n) \\ & - \mathrm{dis}\,(m, n)\,. \end{aligned} \tag{2}$$

λ can be equal to 0, between 0 and 1, equal to 1, or greater than 1.

(i) When no proxy is leveraged, which means mobile node's location is directly exposed, then $\mathrm{dis}(m, p) = 0$ and $\lambda = 0$.

(ii) When proxy is located somewhere between mobile node and peer node, $0 < \lambda < 1$.

(iii) When proxy is located next to peer node, $\lambda = 1$. In this case overhead ϕ is minimized to 0, and λ is 1.

(iv) When proxy is located far away from being between mobile node and peer node, $\lambda > 1$.

Assuming that all proxies a mobile node connects have the same distance $\mathrm{dis}(m, p)$, then the proxies form a circle around the mobile node. $\mathrm{dis}(m, p) = \mathrm{dis}(p, n)$ means peer node happens to be on the circle as well. According to our equation all proxies have the same λ. On the other hand, the proxy which locates at the same location of peer node has minimum ϕ, 0.

Timing is the metric describing how correlated are the inferred and the real movements of the mobile host (i.e., changing network attach point). When no protection mechanism is applied, the adversary can know exactly when the mobile host moves from one location to another. Timing is measured by two types of correlations: number of real network address changes versus exposed network address changes during whole communication and difference of timestamp between real address changes and exposed address changes. The larger the correlation, the better the privacy. Timing privacy must be evaluated for a period of time: during a time range, we assume mobile node moves i times and proxy changes j times. Function $\delta(x, y)$ is used to represent the timestamp difference between events x and y. Then for a mobile node m went through a series of network address change events E^m while its peer node n observed another series of change of proxy server events E^p. Then for a matched

mobile node move E_i^m and proxy server change E_j^p, the timing metric is evaluated as the time difference of these two events:

$$\delta_{E_i^m, E_j^p} = \left| t_i^m - t_j^p \right|, \quad (3)$$

where t is event's timestamp. When evaluating timing privacy we always use the best match of mobile node move events versus proxy server change events, to assume peer node has the best knowledge to leverage that correlation. In another word, we assume the worst case for mobile node is that each of its proxy server changes will be associated with its most recent move. Given that assumption and our δ equation, the overall timing privacy can be evaluated as the sum of

$$\Delta_{m,n,p} = \frac{\sum_{i \in E^m, j \in E^p} \left| \text{time}\,(x, y) \right|}{j}. \quad (4)$$

The higher value of $\Delta_{m,n,p}$, the better timing privacy. The lower bound of $\Delta_{m,n,p}$ is 0 that for each mobile node move, peer can detect its proxy server change at exactly the same time, that is, equivalent to no proxy server. The upper bound is ∞ that when proxy server does not change at all, the exposed address becomes completely static.

Cost. Cost is measured by the operational cost for MSP. Overall MSP needs to pay two parts of operational cost: the cost of running SDN control plane and costs of running dynamic allocated Virtual Routers. The former is relatively static, so it is not considered as optimization goal in this research scope. The latter is determined by the number of proxy servers, their locations (different data center can have different pricing), and bandwidth proxy servers consumed. In a real world deployment, a cloud service provider usually offers various size virtual hosts for different price. However, since the performance-cost ratio of different host type is similar and MSP can find a sweet spot host type, in this research we will assume single host type to simplify the scenario. On mobile node side the cost is associated with the number of proxy server hours it used and bandwidths it consumed. Additionally, mobile node pays higher price for listening proxy server as it represents a more constraint resource, listening port. The operational cost of MSP is

$$C = \sum \text{cost}_{\text{server}} + \text{cost}_{\text{traffic}}. \quad (5)$$

And the cost a mobile node consumes is

$$c_{\text{MN}} = \text{cost}_L * \text{hour}_L + \text{cost}_O * \text{hour}_O + \sum \text{cost}_{\text{traffic}}, \quad (6)$$

where cost is the price of a proxy server per hour use, cost_L is the cost for listening proxy, and cost_O is the cost for outgoing proxy.

Performance. Performance is measured by communication latency and bandwidth, two factors that are the most perceptible for mobile node's experience. Latency is measured by the Round Trip Time (RTT) between mobile node and its peer node, on which proxy location has the most influence. Bandwidth is measured by the highest throughput mobile node can get from proxy node. It is limited by the aggregated traffic bandwidth at a proxy versus proxy node's available physical bandwidth and processing power. In general proxy node has high and stable physical bandwidth and processing power that is large enough to serve a number of Mobile Node's concurrent uses. When proxy node is overloaded mobile node can always be directed to another proxy node in the same data center so bandwidth optimization is based on resource contention and it iss mostly coupled with cost optimization. Latency is then the single parameter of evaluating performance in this research. Latency is only evaluated on live connections between mobile node and peer node via proxy servers. Performance is evaluated by proxy overhead, which is the difference between the RTT directly connected and the RTT connected via proxy.

$$\mu = \text{RTT}\,(m, p) + \text{RTT}\,(p, n) - \text{RTT}\,(m, n). \quad (7)$$

3.7. Protection against Attacks. MSS system enables ubiquitous mobility support while protecting mobile node's privacy. For the four attack models we described the following:

(i) Defending direct connection attack: all connections are indirect and through Proxy. So Bob will never be able to infer Alice's network address while being able to talk with her.

(ii) Defending location registry attack: all location registries only point to Alice's proxy locations. Even if when Bob can acquire multiple proxies' locations, he cannot infer Alice's location since these proxies are set up near to Alice's peer rather than Alice.

(iii) Defending historical location attack: Bob cannot acquire a list of Alice's real network location history either through communication with Alice or through registry.

(iv) Defending location change timing attack: Bob cannot directly detect when Alice changes network location as the connection between him and proxy is always unchanged.

4. Simulation and Discussion

We run simulations to compare MSS versus other models. The comparison is based on the view of individual customer's privacy and performance metrics when one customer moves while communicating with its peers. Besides MSS model, other models are as follows:

(1) Without proxy: this represents an ordinary Internet mobile user always uses direct connections.

(2) Typical VPN user: we simulate by keeping 1 static proxy server selected around the middle of the optimal router between mobile node and its first peer node (within 2 hops). It does give favor to first peer node but statistically it does not make difference if we choose other peer node for simulation.

(3) Typical cell data mobile user: we simulate by keeping a proxy server that follows mobile node move (randomly picked within 2 hops of mobile node's

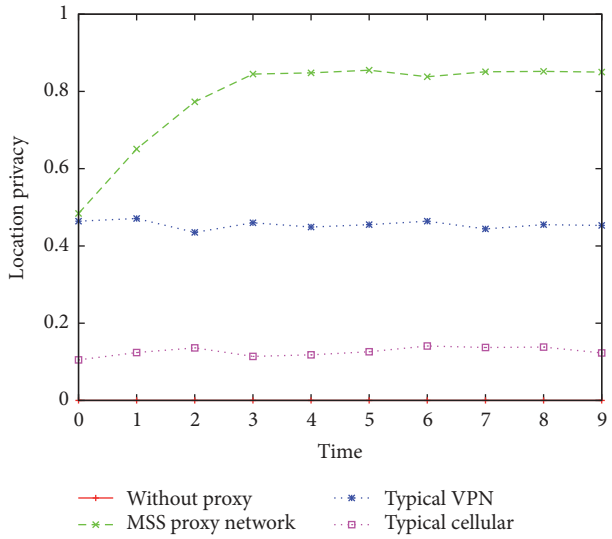

Figure 3: Comparison of location privacy metric for different models.

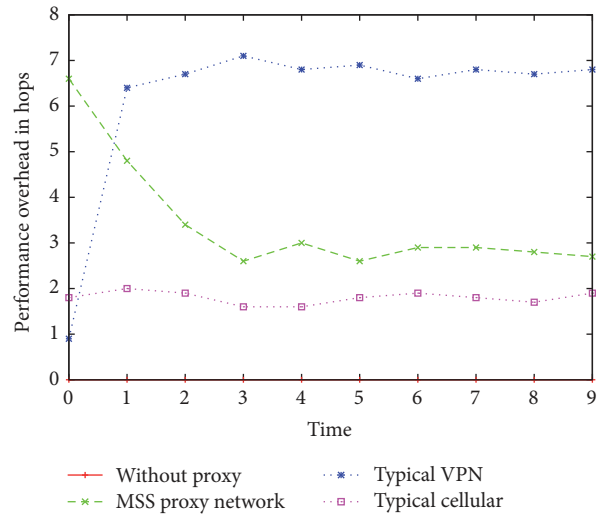

Figure 4: Comparison of performance overhead in hops for different models.

attaching point and reselects when mobile node's access point is moving away from attaching point).

In real world network distance (either hop, latency, or AS distance) is harder to quantify and does not reflect geographical distance well, because network addresses, such as IP, are usually not uniformly distributed or segmented. Fewer network hops do not always mean shorter geographic distance. On the other hand, geographic distance mapped from IP address is easier to quantify and relatively reliable. Even though there are cases that IP address incorrectly mapped to wrong geographical locations (mostly depending on IP database), in reality, this does not impair privacy. Therefore, in our simulation, we will assume all IP addresses can be correctly mapped so our evaluation can rely on geographical distance of network attaching point for comparison.

For simulation, we generate a 1024 node network using BRITE [16] (using BRITE's AS only generation tool and taking AS as a network node) to emulate a segment of Internet topology. Mobile node and peer node are randomly placed on it. The 1024 nodes are randomly distributed in a 2D 1000 by 1000 points plane, and their 2D distance is considered as their geographic distance. Mobile node performs a random walk in 2D plane and connects to the nearest node as its Internet attaching point. During each mobile node movement at ten time spots, we randomly select a peer node talking to it and measure the privacy metric and performance overhead. We generate traces of 100 mobile nodes and compare the average of mobile node's metrics, as shown in Figure 3.

Figure 3 shows location privacy metrics where "no proxy" metric is always 0 as expected. "MSS proxy network" metric quickly increases along time as new proxy is added to proxy network. It is capped around 0.88 as we constrained number of proxy servers by the running cost we artificially set, but the system still is able to optimize performance. "Typical VPN" metric appears to be relatively stable because metrics are averaged out but for individual mobile node metric are spiky. "Typical cellular" metric shows that it does protect location privacy to some extent but is very limited. The real world metric for a cellular user could be worse though cellular network Internet gateway could be closer in terms of geolocation.

Figure 4 shows corresponding performance penalty of using proxy in terms of extra network hops. "No proxy" metric of course is always 0. "MSS proxy network" metric similarly improves quickly for the first few communications and then is limited by the number of proxies. "Typical VPN" metric starts low as our simulation favor first peer node but then quickly goes up. Similarly the averaged metric looks stable but individual mobile node's metric is spiky. "Typical cellular" metric is better than MSS proxy network as we assume cellular Internet gateway is always close by. On the other hand, MSS proxy allocation algorithm still has lots of headroom to optimize and can also leverage prediction from mined historical data.

Due to the implementation of this simulation in which one proxy is static for existing connection for both "MSS proxy network" and "typical VPN," both their timing privacy metric is ∞. "No proxy" metric remains at 0, and "proxy close to mobile node" also remains at 0. The former is expected, but the latter is a little bit stretched where in this simulation each change of proxy server happens exactly the same time when mobile node moves, due to our algorithm and simulation design. However, the difference here is big enough that proxy make mobile node appears to be static, but the other two highly correlate to mobile node's actual moving timestamp.

Figure 5 shows corresponding operation cost in terms of unit time use of proxy. "No proxy" and "typical cellular" metric are always 0 since they did not use any proxy server. "Typical VPN" metric remains at 1 because in our simulation we assume there is always only 1 VPN server. "MSS proxy network" metric shows interesting result though. The cost quickly went up as more proxies are leveraged and finally capped around 3.8 as in our simulation we

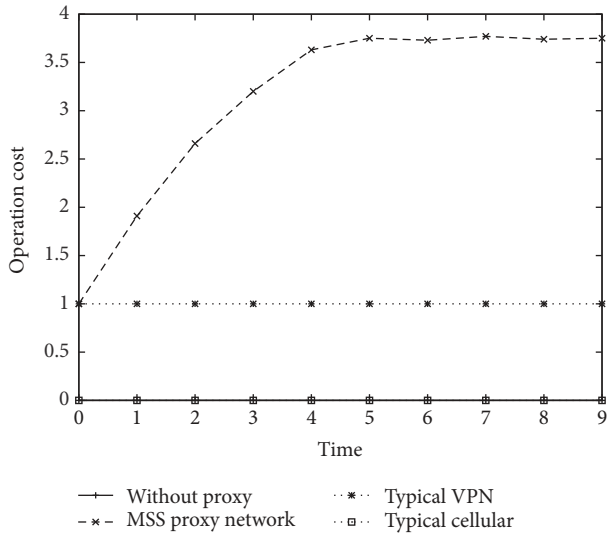

Figure 5: Comparison of operation cost for different models.

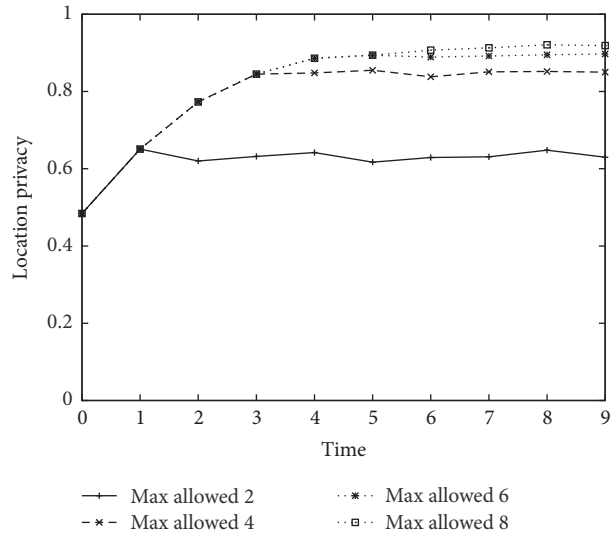

Figure 6: Comparison of location privacy metric for different allowed max proxies.

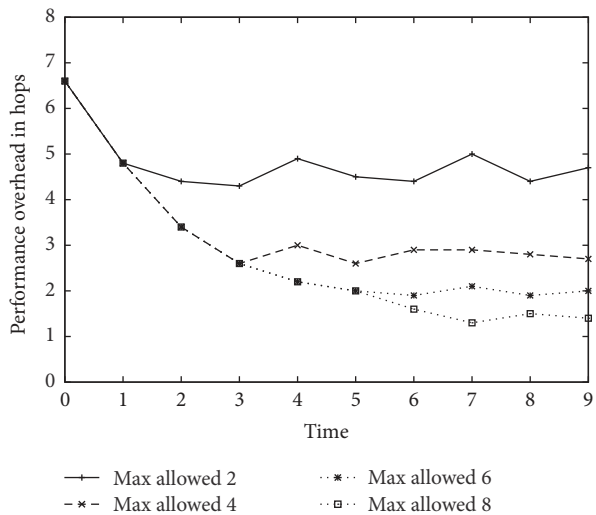

Figure 7: Comparison of performance overhead in hops for different allowed max proxies.

set the upper bound of proxy allocation is 4. During the same time from Figure 3 we can see privacy metric also correspondingly quickly went up as those new proxies were likely to be allocated at relative "edge" location and therefore improved distance privacy greatly. Performance was also quickly optimized along with increase of proxies, although there was an interesting churn that happened at time 4. Based on analysis our theory is that because the call pattern we generated is uniformly random, a large portion of new allocated proxies at time 3 does not serve request at time 4 well as they were allocated away from both mobile node and peer nodes. After that proxy allocation is then adjusted to adapt; the performance overhead went down again and remained relatively stable.

Another group of simulation we performed is a comparison between the different maximum allowed Proxies; that is, we allowed maximum 2, 4, 6, and 8 proxies for each mobile node in simulation and compared result as shown in Figures 6 and 7. Figures 6 and 7 both show that when allowing more proxies both performance and privacy improved. On the other hand, the percentage of improvement starts to decrease when allowed maximum proxies are more than 4 and became less significant when allowed proxies increased to 8.

When looking at the corresponding operation cost from Figure 8 we can see that the number of proxies increased gradually. When more proxies were created, allocation of new proxies became less aggressive. For the simulation iteration allowing 8 maximum proxies, only 84.6% mobile nodes had 8 proxies setup at the end. This suggests that when there is enough number of proxies adding more does not necessarily mean better performance and privacy, at least for the given simulation setup and algorithm implementation. Actually for the simulation iterations of having 6 and 8 proxies, mobile node's cost/performance ratio dropped compared to iterations having 4 proxies. In other words the simulation result also suggests that having 4 maximum proxies is the sweet spot, balancing the operation cost, performance overhead, and privacy.

5. Related Works

Privacy issue caused by mobility, especially location, has been well studied. de Montjoye et al. [17] found that from a large set of anonymous movement data, using four data points of hourly data can identify 95% unique users. Given the identified movement pattern of this identified user, they can even construct a history of this user's locations from the anonymized data set. Ma et al. reached the same conclusion [7]. This clearly shows location information, even after anonymization, can greatly threaten a mobile user's privacy when it can be collected by the adversary. Cloud

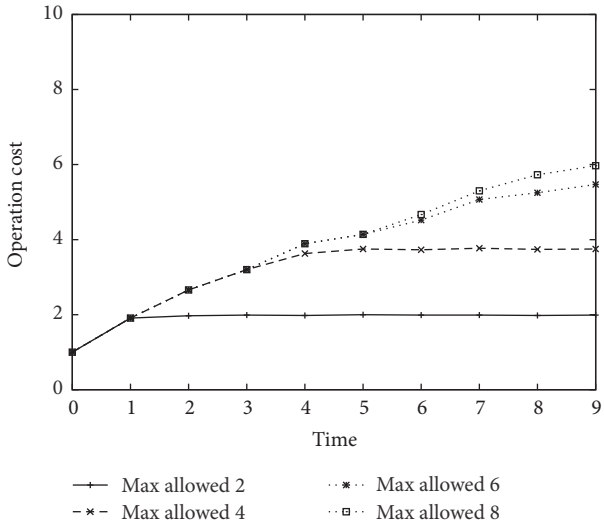

Figure 8: Comparison of operation cost for different allowed max proxies.

computing and big data just make this exploit more available and accessible [18]. However, without proper protection, a mobile host cannot hide its location since its network location will be exposed to any peer it communicates and the network location is an approximation of its physical location.

One major research area is to protect against user profiling from Location Based Service (LBS) while still being able to use it [19–21]. Wernke et al. [22] surveyed different identity protection types and common mechanisms to protect and attack privacy. Usually, a compromise between quality of service and privacy is optimized by manipulating location reporting frequency, precision, or both [23]. For example, Shokri et al. [24, 25] designed an approach to hide user's profile against adversary by solving it as Bayesian Stackelberg game. Primault et al. proposed mechanism to reduce profiling exposure, by hiding POI where the user stops, and also let user exchange their trajectories when meeting [26]. On the other hand, research work also shows obfuscated location data and can improve location privacy but cannot stop adversary to infer relative precise Point Of Interests (POI) [27].

Another focusing area is to increase anonymity of collected user location data [11], limiting shared location information, or evaluating privacy exposure level before sharing location data [12]. On the other hand, there are research works pointing out that because human mobility trace is very unique [17], even completed anatomized data can still be used to extract patterns and identify individuals [7–9]. So as long as relative location and movement are collected, location privacy can be compromised to certain context.

MobilityFirst [28, 29] is a proposed new Internet architecture that emphasizes mobility support, comparing to existing Internet architecture. Privacy and communication security are one major challenge for MobilityFirst. Access control is proposed to be applied to MobilityFirst so that only allowed network entities can contact a host or resolve its network locations [30].

Research works done show that people care about their location privacy but also are willing to trade it off for financial benefit, for example, lowering car insurance payment by exposing location to insurer [31].

6. Conclusions

In this paper, we discussed privacy issues a communication connection of Internet mobile user could expose and corresponding attack model. Based on that we proposed a design to protect it with a dynamic distributed proxy network, built upon our previously designed Mobility Support System. An Internet user can receive ubiquitous Internet mobility support while network location privacy being well protected, with a marginal performance overhead and minimized operation cost. Our preliminary simulation results show that our design meets our expectation, and there is still headroom to optimize for performance and cost in the future.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] J. Laganier, T. Koponen, and L. Eggert, "Host Identity Protocol (HIP) Registration Extension," RFC Editor RFC5203, 2008.

[2] M. Buddhikot, A. Hari, K. Singh, and S. Miller, "MobileNAT: A new technique for mobility across heterogeneous address spaces," *Mobile Networks and Applications*, vol. 10, no. 3, pp. 289–302, 2005.

[3] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "The Locator/ID Separation Protocol (LISP)," RFC Editor RFC6830, 2013.

[4] Apple push notification service.

[5] Google cloud messaging.

[6] H. Lee and E. Chuvyrov, *Beginning Windows Phone 7 Development*, Apress, Berkeley, Calif, USA, 2010.

[7] C. Y. T. Ma, D. K. Y. Yau, N. K. Yip, and N. S. V. Rao, "Privacy vulnerability of published anonymous mobility traces," *IEEE/ACM Transactions on Networking*, vol. 21, no. 3, pp. 720–733, 2013.

[8] M. Lin, H. Cao, V. Zheng, K. C. Chang, and S. Krishnaswamy, "Mobile user verification/identification using statistical mobility profile," in *Proceedings of 2015 International Conference on Big Data and Smart Computing, BIGCOMP 2015*, pp. 15–18, February 2015.

[9] G. Tsoukaneri, G. Theodorakopoulos, H. Leather, and M. K. Marina, "On the inference of user paths from anonymized mobility data," in *Proceedings of the 1st IEEE European Symposium on Security and Privacy*, pp. 199–213, Saarbrucken, Germany, 2016.

[10] V. Kulkarni, A. Moro, and B. Garbinato, "A mobility prediction system leveraging realtime location data streams: poster," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking, MobiCom '16*, pp. 430–432, New York, NY, USA, 2016.

[11] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: architecture and algorithms," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 1–18, 2008.

[12] I. Boutsis and V. Kalogeraki, "Location privacy for crowdsourcing applications," in *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '16*, pp. 694–705, York, NY, USA, 2016.

[13] P. Zhang, A. Durresi, and R. Jain, "Cloud aided internet mobility," in *Proceedings of the IEEE International Conference on Communications, ICC 2013*, pp. 3688–3693, Budapest, Hungary, 2013.

[14] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *Proceedings of the IEEE Symposium on Security and Privacy, SP 2011*, pp. 247–262, Berkeley, Calif, USA, 2011.

[15] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: differential privacy for location-based systems," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '13)*, pp. 901–914, ACM, Berlin, Germany, 2013.

[16] Alberto Medina, Lakhina Anukool, Ibrahim Matta, Byers John, and Brite, "Universal topology generation from a user's perspective," Tech. Rep., Boston, Mass, USA, 2001.

[17] Y.-A. de Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel, "Unique in the crowd: the privacy bounds of human mobility," *Scientific Reports*, vol. 3, article 1376, 2013.

[18] R. Lu, H. Zhu, X. Liu, J. Liu, and J. Shao, "Toward efficient and privacy-preserving computing in big data era," *IEEE Network*, vol. 28, no. 4, pp. 46–50, 2014.

[19] K. G. Shin, X. Ju, Z. Chen, and X. Hu, "Privacy protection for users of location-based services," *IEEE Wireless Communications*, vol. 19, no. 1, pp. 30–39, 2012.

[20] Maria Luisa Damiani, "Privacy Enhancing Techniques for the Protection of Mobility Patterns in LBS: Research Issues and Trends," in *European Data Protection: Coming of Age*, Chapter 10, pp. 223–239, Springer, Dordrecht, Netherlands, 2013.

[21] N. Pelekis and Y. Theodoridis, "Mobility data management and exploration," *Mobility Data Management and Exploration*, pp. 1–300, 2014.

[22] M. Wernke, P. Skvortsov, F. Dürr, and K. Rothermel, "A classification of location privacy attacks and approaches," *Personal and Ubiquitous Computing*, vol. 18, no. 1, pp. 163–175, 2014.

[23] B. Niu, X. Zhu, W. Li, H. Li, Y. Wang, and Z. Lu, "A personalized two-tier cloaking scheme for privacy-aware location-based services," in *Proceedings of 2015 International Conference on Computing, Networking and Communications, ICNC 2015*, pp. 94–98, Garden Grove, Calif, USA, 2015.

[24] R. Shokri, G. Theodorakopoulos, and C. Troncoso, "Protecting location privacy: optimal strategy against localization attacks," in *Proceedings of the 19th Conference on Computer and Communications Security (CCS '12)*, pp. 617–627, Raleigh, NC, USA, 2012.

[25] G. Theodorakopoulos, R. Shokri, C. Troncoso, J.-P. Hubaux, and J.-Y. Le Boudec, "Prolonging the hide-and-seek game: optimal trajectory privacy for location-based services," in *Proceedings of the 13th Workshop on Privacy in the Electronic Society, WPES 2014, in Conjunction with the ACM Conference on Computer and Communications Security, ACM CCS 2014*, pp. 73–82, Scottsdale, Ariz, USA.

[26] V. Primault, S. B. Mokhtar, and L. Brunie, "Privacy-preserving publication of mobility data with high utility," in *Proceedings of 35th IEEE International Conference on Distributed Computing Systems, ICDCS 2015*, pp. 802-803, Columbus, Ohio, USA, 2015.

[27] Vincent Primault, Sonia Ben Mokhtar, Cédric Lauradoux, and Lionel Brunie, "Differentially private location privacy in practice," *Computing Research Repository*, 2014, abs/1410.7744.

[28] "Mobilityfirst future internet architecture project," http://mobilityfirst.winlab.rutgers.edu/.

[29] F. Bronzino, K. Nagaraja, I. Seskar, and D. Raychaudhuri, "Network service abstractions for a mobility-centric future internet architecture," in *Proceedings of the 8th ACM international workshop on Mobility in the evolving internet architecture, MobiArch '13*, pp. 5–10, Miami, Fla, USA, 2013.

[30] Xiruo Liu, *Integrating security and privacy protection into a mobility-centric internet architecture, [Ph.D. thesis]*, Rutgers, The State University of New Jersey, 2016.

[31] S. Derikx, M. De Reuver, M. Kroesen, and H. Bouwman, "Buying-off privacy concerns for mobility services in the Internet-of-things era: a discrete choice experiment on the case of mobile insurance," in *Proceedings of 28th Bled eConference, #eWellBeing*, pp. 228–238, Bled, Slovenia, 2015.