Where the Sidewalk Ends: Privacy of Opportunistic Backhaul

Tess Despres
tdespres@berkeley.edu
University of California, Berkeley

Shishir Patil
shishirpatil@berkeley.edu
University of California, Berkeley

Alvin Tan
alverino@berkeley.edu
University of California, Berkeley

Jean-Luc Watson
jlw@berkeley.edu
University of California, Berkeley

Prabal Dutta
prabal@berkeley.edu
University of California, Berkeley

Abstract

We explore the challenges of implementing a privacy-preserving opportunistic data collection network for resource-constrained edge IoT devices. Opportunistic networks require no fixed infrastructure and allow edge devices to piggy-back messages through stationary or mobile gateways. Research interest in such networks has waxed and waned over the years, but commercial deployments have not taken off until recently. Over the past year, we have witnessed a resurgence of interest fueled by wide-scale commercial deployments, most notably Amazon's Sidewalk network, but also Apple's *Find My* and the Tile network. As these networks become more prevalent, maintaining the privacy of the individuals who participate in them will become increasingly important. In this paper, we demonstrate that current opportunistic networks leak access patterns to the network operator itself through communication metadata, which can be used to reconstruct location traces. We argue that opportunistic networks and privacy are not mutually exclusive, and suggest some potential research directions to strengthen the privacy properties of these networks. Since opportunistic networks are now being deployed at massive scale, we argue that the time is ripe to make them privacy-preserving before it is too late.

ACM Reference Format:
Tess Despres, Shishir Patil, Alvin Tan, Jean-Luc Watson, and Prabal Dutta. 2022. Where the Sidewalk Ends: Privacy of Opportunistic Backhaul. In *15th European Workshop on Systems Security (EUROSEC'22), April 5–8, 2022, RENNES, France.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3517208.3523757

1 Introduction

Researchers have been working towards the ubiquitous deployment of mobile sensor devices for decades [42]. However, the success of ubiquitous sensing applications has historically been limited by the need to have an associated wide area backhaul infrastructure. Opportunistic mesh protocols [10, 34, 41] have sought to minimize this burden by creating a mesh network among device nodes to expand connectivity past the range of Internet-connected base stations. Unfortunately, this still requires a base station deployment at scale and relies on deployed devices to encounter each other at frequent intervals. The last few years have seen commercial efforts to convert third-party platforms like mobile phones, or other already-deployed infrastructure such as smart home devices, into gateways that can vastly expand network range without relying on a mesh-like structure. Gateways are pre-existing, often owned by third-parties, and may be mobile, so their locations cannot be planned with pin-point accuracy. However, expanding the number of available gateways means that coverage is present at such a large scale that edge devices are likely to pass within connection range frequently. This enables practical applications – asset tracking [1, 39], urban sensing [5], health monitoring [18], or wildfire tracking [7] – without additional infrastructure deployment cost.

Despite the expansion of opportunistic backhaul networks to a broader category of gateways, until recently, systems such as Apple's *Find My* [1] and the Tile [39] location-tracking network have remained vertically integrated, relying on a network of first-party gateways. These systems highlight a major issue: any new large-scale deployment would have to source their own backhaul mechanism, by either placing enough devices to ensure coverage or incentivizing consumers to host a gateway application. As a result, commercial efforts are increasingly providing opportunistic backhaul as a service. Google's Physical Web connected Android phones to nearby devices until it shutdown in 2018 due to a high amount of spam [32], Comcast routers broadcast WiFi hotspots to nearby subscribed users [6], the Helium network incentivizes users to deploy LoRa gateways [17], and Amazon recently launched Sidewalk [3], an opt-out BLE network that provides backhaul through residential Internet connections.

We believe that opportunistic network infrastructure is at a similar inflection point today as peer-to-peer networking was during the widespread deployment of Napster and BitTorrent. Whereas backhaul was previously deployed in the context of academic projects, or deeply integrated into a particular application, centralized providers will enable backhaul services for any mobile device deployment at an unprecedented scale and density. Unfortunately, coalescing responsibility for backhaul routing into a small number of entities (e.g. Apple or Amazon) has significant privacy implications. While some public areas such as shopping centers have used WiFi connections to track customers during their visit [16], widely deployed networks like Sidewalk have the potential to silently track individuals throughout their day with almost no interruption.

Importantly, current solutions for securing wireless protocols are not enough to protect user privacy from the backhaul network operators themselves. Wireless MAC address rotation is a standard

EUROSEC'22, April 5–8, 2022, RENNES, France

ACM ISBN 978-1-4503-9255-6/22/04.
https://doi.org/10.1145/3517208.3523757

feature designed to prevent 3rd party snoopers from tracking devices by a long-term WiFi or Bluetooth address [2], but devices are prone to leaking long-term identifiers anyway [29, 30] and it does nothing against backhaul networks that have knowledge of long-term device identifiers. Further, while Apple's *Find My* and the Apple/Google Exposure Notifications project [8] integrate major protections for user privacy, these very changes make them are unsuited to private data backhaul. *Find My* requires user authentication with Apple to both write and read from its database, allowing the company to infer social connections between different users [20], and the need to route data from gateway devices to a desired destination device would only strengthen these links. Similarly, exposure notifications preserve privacy by keeping "seen" identifiers locally on-device. Using user devices to backhaul data packets, however, would link them at the server level breaking the protocol's privacy guarantees.

The natural approach to implementing a centralized backhaul system yields an undesirable privacy outcome. In particular, efforts to solve practical deployment issues like spam prevention or device authentication yield detailed metadata, including device and gateway identifiers. This data allows the backhaul service provider to uniquely identify and follow participants even when they cannot directly inspect application payloads. We expect mobile devices, already very capable cellular, WiFi, and BLE-equipped platforms, to be folded into backhaul deployments, significantly increasing the impact of metadata accumulation on individual privacy.

The notion that location data can be used to reconstruct a large amount of personal information is not new. Given access to a user's personal *mobility trace*, their identity, home address, work location, or political views can be easily inferred [9]. Further, Shen et al. [35] demonstrated that it is possible for centralized cellular network providers to reconstruct a user's mobility trace efficiently given ground-truth cell tower coordinates. This paper seeks to confirm that routing metadata allows an opportunistic network operator to recreate participant mobility traces.

Due to its timeliness and scale, we focus on the specification for Sidewalk [3] as a representative centralized backhaul network with support for third-party applications. We simulate a deployment over real-world mobility data (Figure 1). We first confirm that with knowledge of deployed gateway locations, the density envisioned by opportunistic backhaul allows for precise mobility trace reconstruction. We then show that this prior knowledge is not strictly necessary: limited only to the metadata collected from payload routing and a small number of placed gateways, we reconstruct the positions of other gateways in the network. This result supports our position that routing metadata in backhaul networks puts user mobility privacy at risk.

However, privacy does not need to be sacrificed to realize opportunistic systems. We outline a number of promising research directions to enhance the privacy-preserving properties of backhaul systems, which could allow a service provider to route application payloads without leaking device movement patterns. In the end, every privacy-conscious choice requires a trade off in system complexity or overall capability — we discuss the benefits and drawbacks of protecting user behavior, with an eye towards enabling privacy as a first-class feature in the next generation of deployments.

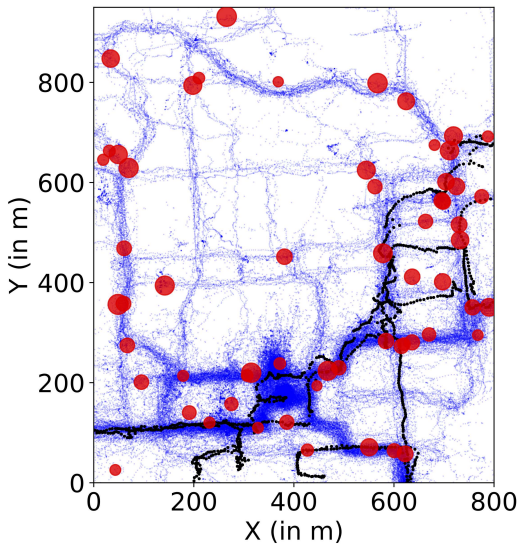

Figure 1: Ground truth device location traces (blue) in the GeoLife dataset [43] and simulated gateway positions (red) on Peking University campus. The black path highlights a specific mobility trace we attempt to reconstruct, discussed in Section 3.

2 Background and Related Work

In this section, we first identify how recent commercial designs to expand edge network connectivity differ from existing academic and application-specific deployments. We then discuss Amazon's Sidewalk deployment, and detail how derived location data could seriously compromise client privacy.

Opportunistic mesh networks. Much prior work focuses on mobile mesh architectures for data backhaul. Sensors, such as the MULEs (Mobile Ubiquitous LAN Extensions) introduced by Shah et al. [34], transmit packets over short-range wireless links with peers until they reach a base station; this area has yielded a robust body of protocol-based work [27, 31, 36, 37, 41]. On the security side, encounter-based communication protocols [25, 28, 40] rely on physical proximity to generate secure keys for inter-device communication. Unfortunately, avoiding device identifiers in these networks opens the door to spammers and passive eavesdroppers.

Deployed mesh-based systems do not provide strong privacy guarantees for participants that backhaul application data. In the case of *Find My*, Apple devices scan for BLE advertisements to crowd-source an approximate location for another lost device, which is reported back to the device owner. However, because Apple's servers authenticate devices proxying location reports to the cloud, they can easily correlate user locations based on the time and uploader identity [20]. Exposure Notifications [8] work in a similar manner by having individual devices retain Bluetooth Low Energy (BLE) beacons seen from other nearby users to later identify any interactions that could have exposed them to COVID-19. In this case, privacy is protected by keeping observed beacon packets locally without uploading them to the cloud, which would not be possible for the device to avoid when backhauling payloads.

Backhaul as a service. To provide edge connectivity at scale, the most recent backhaul infrastructure systems have supported

third-party deployments by focusing on a gateway-centric design. Just like WiFi or cellular-based networks for more powerful consumer devices, these systems provide gateways to proxy data received over BLE or another low power wireless protocol. Crucially, many such services can be scaled using existing hardware. Google's Physical Web aimed to facilitate interactions with nearby devices, operating on nearly every Android smartphone between June 2016 and its eventual shutdown in December 2018. Adkins et al. [4] demonstrated how web page fetching behavior could be leveraged into an opportunistic backhaul mechanism. LoRA-based [33] networks enable up to 8 km of communication range, but allow limited bandwidth, require radio line-of-sight, and can be bottlenecked by the quantity of deployed gateways. MachineQ [21] provides a fully-centralized LoRaWAN network, while Helium [17] uses a decentralized ledger to maintain gateways, provide location services, and ensure payment for data backhaul.

Sidewalk. Recently deployed in June 2021, Amazon's Sidewalk [3] system operates on all Amazon devices (e.g. Amazon Echo, Ring cameras, etc.) and is turned on by default. The network represents a continuation of the trend towards large, centralized gateway deployments, and has significant potential reach, supporting both BLE and 900 MHz (e.g. LoRa) wireless communication. Third-parties can use these gateways to offload data through BLE as they enter the gateway's range, which is relayed to the relevant destination server through a centralized routing service. To deliver application data and enable bi-directional communication, Sidewalk collects *routing metadata* at a central network server for each payload. Specifically, Sidewalk (1) authenticates the gateway being used and records recently-used gateways for bidirectional communication, (2) collects endpoint identifiers to authenticate devices, (3) keeps gateways time-synchronized to generate correct payload timestamps, and (4) is given the desired server destination for the application data. Unfortunately, while several encryption layers and rotating transmission identifiers protect Sidewalk communication, no guarantees can be made on how Amazon itself handles user metadata. The Sidewalk security analysis [3] relies only on a (self-enforced) data retention policy to periodically wipe out routing metadata. For example, the system claims to forget the device ID associated with a transmission after replacing it with a temporary rotating identifier. In reality, the same analysis details how device IDs are kept to enable bidirectional communication, as the most likely gateway to still be in communication with the device is the one that handled its last transmission. In the end, users must place full trust in Sidewalk to deliver on their data management policies with no effective guarantee of privacy built into the system design itself. Companies may easily change policies or deceive customers [22] to continue collecting data while maintaining the public perception of privacy.

Breaching user privacy with mobility traces. Knowledge of a person's movement patterns represents a substantial breach of privacy. De Montjoye et al. [13] showed that a majority of mobility traces with very low cardinality, containing as little as 4 datapoints, could be uniquely tied to a particular person. These can then be combined with external information (e.g. estimated home and work locations from public records) to deanonymize the trace owner. Srivatsa and Hicks [38] demonstrated this process, using a social network graph to unmask users based on how often their mobility traces intercepted each other. Even indirect location sharing, based on connections to other parties in a social network, has been used to recover mobility traces [26]. Once identified, the lack of location privacy leaks a wide array of sensitive information based on visited locations: home addresses [19], political leanings from attending campaign rallies, medical procedures based on visited clinics, or job searches requiring interviews at competing firms [9].

3 Mobility Trace Reconstruction

To illustrate the privacy implications of large scale opportunistic backhaul deployments, we design and evaluate a proof-of-concept mobility trace reconstruction, using simulated routing metadata. We simulate a Sidewalk-like deployment where the device ID, gateway ID, and transmission time is collected for each connection and retained. We demonstrate how a user's mobility trace can be closely tracked given knowledge of gateway locations. In the absence of ground truth locations for all but a few gateways, we demonstrate how location information can still be estimated from timestamped connection sequences and used to recreate mobility traces.

3.1 Setup

We use Microsoft's publicly-available GeoLife mobility dataset [43] to simulate pedestrian mobility. The dataset is collected by 182 users in a period of over three years ending in August 2012, with 91% of the trajectories logged in a dense representation, e.g. every 1∼5 seconds or every 5∼10 meters per point.

We simulate a set of pedestrians carrying network-enabled endpoint devices while moving around the campus of Peking University in Beijing, China over a 800 by 950 meter area. We also simulate a random deployment of opportunistic gateways along pedestrian routes, where they encounter the passersby carrying mobile endpoints. In total, we simulate 76 stationary gateways and parse out over 1000 mobility traces from the GeoLife dataset, such that each mobility trace follows a pedestrian's movement over 5∼15 minutes. The traces and the locations of the gateways are shown in Figure 1.

We divide our analysis into location-based and metadata-based reconstruction. In the former, we predict the movement of an endpoint over time by tracking the gateways that the device opportunistically connects with. We assume the backhaul network provider has preexisting knowledge of where each gateway is, and can reconstruct the device's movement by interpolating between gateways. This process has been seen in prior work (e.g. [13, 35]), but the underlying assumption is very strong. A backhaul network provider may not know a gateway's location, especially when operating on third-party hardware. The second variant of our analysis demonstrates that even without knowledge of many gateway locations, a network provider can leverage its collected routing metadata and the locations of a few known gateways to estimate the other unknown gateway positions. The mobility trace for a particular device can then be estimated using location-based reconstruction.

Mobile devices are represented by GeoLife trajectories in our target area, and move at a velocity matching their recorded GPS coordinates. Stationary gateways are randomly distributed across the campus based on mobility density: where more people travel, more gateways are deployed. All devices broadcast twice per second, with simulated gateway ranges uniformly distributed between 10 and 20 meters. If a device broadcasts within range of a gateway, a connection is logged containing the metadata discussed in Section 2.

We assume that the backhaul network provider acts in an *honest but curious* fashion, gathering a persistent history of transmission metadata (device and gateway identities, and transmission time), but does not collude with other transmitting devices in the network or actively prevent application payloads from being routed.

3.2 Location-based reconstruction

We first assume that all backhaul gateway locations are known to the network provider. As devices move around and connect to gateways, the network provider collects the connection metadata to trace a sequence of visited gateways, thus roughly extracting a device's location at the time of transmission. Given sufficient position observations, as the result of connecting to a gateway, we can reconstruct an accurate mobility trace.

We demonstrate this process by focusing on a single mobile device, whose trajectory and visited gateways over the course of an hour are shown as the black overlay trace in Figure 1. We then using linear splines, an interpolation function defined piece-wise by polynomials, between gateway positions to reconstruct the device's movement over time. The reconstructed mobility trace and actual device position over time are shown in Figure 2. In general, the accuracy of the reconstructed trajectory increases as more connection events are observed. When gateways are sparse, such as between 800 and 2200 seconds into the mobility trace, the spline estimate is oblivious to any detours the device might make. For some of the other endpoints we considered, this sparsity in gateway information caused our spline-based reconstruction to stray up to 400 meters from the ground truth. However, for the relatively well-covered trace in Figure 2, our spline-based reconstruction stays within 45 meters on average from the ground truth device position. This demonstrates that as commercial deployments allow greater coverage, mobility trace reconstruction become more precise.

3.3 Metadata-based reconstruction

We detail a reconstruction method that would allow an adversarial network provider to recover device mobility traces using only sparse gateway location information. In this scenario, network providers have flexibility in that they only have to deploy a few gateways at known locations with high traffic flow.

Specifically, by pairing the known locations of a few gateways with the connection sequences generated by devices moving through the area, we estimate the positions of other nearby gateways through triangulation. Using these estimated positions, we can then reconstruct the movement of devices through an area, even if a device never connected to any of the gateways with known positions. Thus, not only can an adversarial network provider reconstruct the movement of endpoints through that area over time, but they can also derive an estimated position for the other gateways.

3.3.1 Estimating pairwise distances For each of the devices in our sample traces, p_i for $i \in \{0, ..., 1034\}$, we have a sequence of connections with the gateways g_j and the times t_k they occurred: $(g_j, t_k)_i$ for $j \in \{0, ..., 75\}$ and $t_k \in [0, \tau_i]$ for total trajectory times $\tau_i \in [5, 15]$ minutes. Given this metadata, we can estimate the symmetric matrix $D \in \mathbb{R}^{76\times76}$ of pairwise distances between gateways in the area. Specifically, for each trace p_i, we calculate the list of time differences $(t_{k_1} - t_{k_2})$ between connections made with gateways g_{j_1}, g_{j_2} for connection times t_{k_1} and t_{k_2} that occurred within

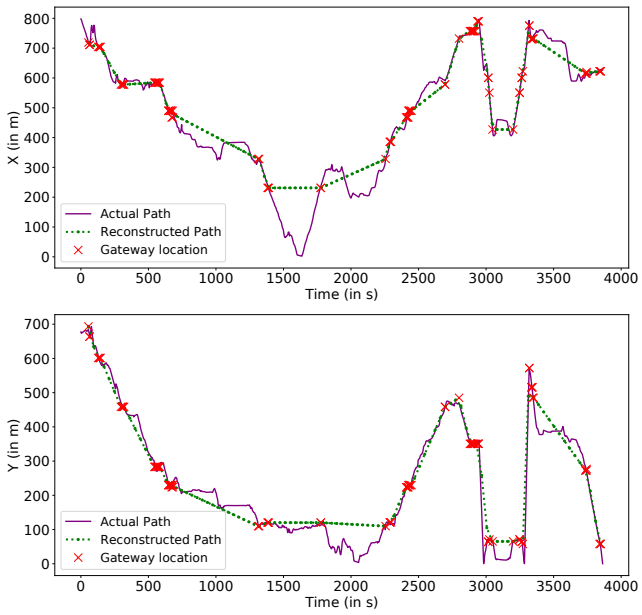

Figure 2: Mobility trace reconstruction in the X- and Y-coordinates with ground truth gateway locations. The purple curve indicates the ground-truth path of the sensor, the green spline indicates the trace we predicted from our attack, and each red X indicates the location of a gateway the sensor encountered.

two minutes of each other. Since we want an accurate straight-line distance between gateways in order to conduct triangulation, we select the $5th$ percentile value of $(t_{k_1} - t_{k_2})$ for each pair of gateways to use as the time distance estimate, avoiding noise. This gives us a symmetric matrix $T_i \in \mathbb{R}^{76\times76}$ of pairwise time differences between area gateways for each trace, sparse (due to the locality of the traces) matrix. We ignore any trace that does not see at least three unique gateways, as traces with only two or less gateways do not provide any meaningful information about relative distance between gateways. Of the 1034 traces we started with, only 637 of them passed by at least three unique gateways, with the other 397 traces being too short or walking in too sparsely populated areas to interact with enough gateways.

To convert these time differences T_i into physical distances, we make the simplifying assumption that each endpoint moves at a constant speed v_i throughout its trace. Our data validates our assumption - standard deviation of the velocities of the endpoints we used tend to be around 1 m/s. We estimate the speed of each endpoint by taking the pairwise distances between known gateways and dividing it by the pairwise time differences between them (i.e. $v_i = mean\{dist(g_{j_1}, g_{j_2})/T_i[j_1, j_2]\}$ for $j_1, j_2 \in known \subseteq \{0, ..., 75\}\}$). For reconstruction, we selected 9 of the 76 gateways as our *known* gateways, specially picking gateways that were well-spaced in frequently traveled areas. We then calculate D_i for speed v_i from the matrix T_i using the proportional relationship $D_i = v_i T_i$. We take the median values of these D_i matrices to get our aggregated D matrix. Of the 637 T_i matrices containing at least three gateways, only 42 of them contained a pairwise time difference between two known gateways. Since the majority of paths did not

	Min. device interactions	Gateways used	Gateway localization error (m)	Reconstructed mobility trace duration (h)	Avg. mobility trace error (m)
Ground truth	-	76	-	38	19 ± 24
Metadata-based estimate	1	50	136 ± 188	31	79 ± 115
	3	36	51 ± 97	25	44 ± 57
	5	31	35 ± 57	23	42 ± 56

Table 1: Comparison of mobility trace reconstruction amount and accuracy depending on which gateways are used in reconstruction. The ground truth baseline assumes perfect knowledge of all 76 gateway locations and is able to estimate a cumulative 38 hours worth of mobility traces with an average error of 19 meters. For metadata-based estimates, relying on gateways with more precise location estimates lowers the quantity of paths reconstructed but achieves higher reconstructed path accuracy.

pass by known gateways, we used our calculated D matrix to estimate the speed of other sensors that have time estimates for the path between at least 3 gateways pairs present in D. This allows us to extrapolate positions for additional gateways. We incorporate 15 more T_i matrices for a final pairwise distance matrix D built using gateway-device interactions from 57 unique traces.

It is worth noting the value of collecting many mobility traces and of carefully selecting the locations to place known gateways. Although we started with over 1000 mobility traces, we were only able to convert 57 of them into useful distance information due to the positioning of our known gateways and the density and position of other gateways in the area. Adjusting the positions of our known gateways or using a different number of known gateways will likely affect how many traces can be used in our distance estimates, but this is auxiliary to the point we make. We simply note that known gateway locations should be chosen intelligently, and more mobility data allows for more accurate reconstructions.

3.3.2 Triangulating positions of other gateways Once we have constructed D, our distance estimates between gateway pairs, estimating the location of each gateway becomes an optimization problem.

Specifically, we solve the following:

$$\min_{pos(g_{j_u})} \sum_{j \in \{0,\ldots,75\}} (||pos(g_{j_u}) - pos(g_j)||_2 - D[j_u, j])^2$$

for each $j_u \in unknown = \{0, ..., 75\} \backslash known$, where $unknown$ represents gateways with unknown locations and $pos(g)$ is the (x, y)-position of gateway g. We minimize the difference between the distances between the predicted positions and the values in D to estimate $pos(g)$ for each gateway. We do this through iterative least squares optimizations on $unknown$ gateways until the positions stabilize. To avoid local minima, we instantiate the predicted position values randomly, run 20 predictions with randomized initial positions, and select predictions that minimize the loss. This process gives us gateway position estimates $pos(g_{j_u})$ for $j_u \in unknown$.

3.3.3 Results We find that the accuracy of this reconstruction depends on the number of usable traces that pass by each specific gateway, and we explore this relation in Figure 3. If a usable trace p_i passes by a specific gateway and generates useful values in its T_i matrix, we flag a *device interaction* for that particular gateway. We count up these interactions across all 57 of our useful traces to get the total number of device interactions for each gateway. We then consider the Euclidean errors of the gateway predictions based on how many device interactions each gateway has.

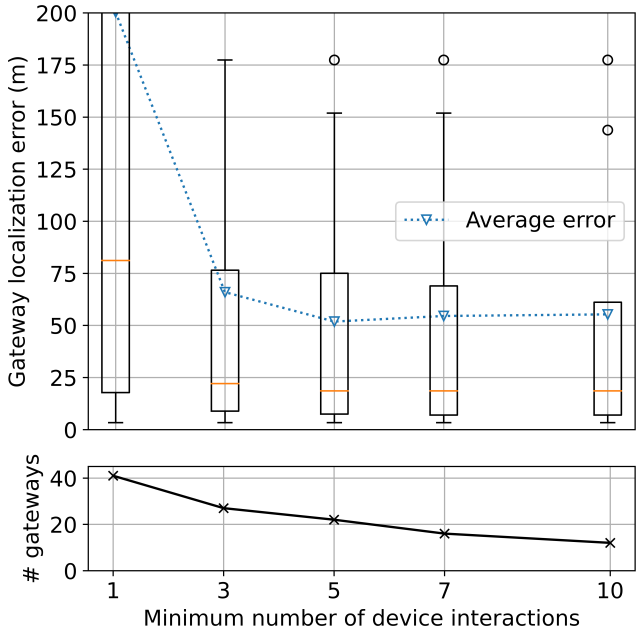

Figure 3: Gateway localization errors of unknown gateways versus the number of device interactions recorded for each gateway. With at least five device interactions, we are able to estimate gateway locations with an average error of 50m and a median error of 20m. Given that the simulated BLE ranges is 10-20m, these are reasonably good estimates. Thus, when 9 known gateways are intelligently deployed, a network provider could locate over 20 other personally-owned gateways to within 50m on average, the size of a small city block. Such estimates are multiple orders of magnitude more accurate than localization based only on gateway IP addresses.

The 57 traces cumulatively interact with around 40 of the 75 gateways present, but many of these 40 gateways only have one or two device interactions, resulting in an average gateway prediction error of over 150 meters. However, if we only consider the ~20 gateways that have at least 5 device interactions, the location estimate quickly improves to have an average error of around 50 meters and a median error under 25 meters. Thus, while it may be difficult to collect usable mobility traces, it only takes a few useful traces to get a reasonable position estimate for each gateway.

We use our predicted gateway positions to carry out the localization attack from Section 3.2 on all the traces in the Geolife

dataset and compare the aggregate results against a ground truth baseline in Table 1. The baseline assumes perfect knowledge of all 76 gateway locations and estimates a cumulative 38 hours worth of mobility traces with an average error of 19 meters. For each metadata-based estimate, we use the set of gateways with at least *Min. device interactions* number of interactions in addition to our 9 known gateways to generate trace predictions. Since we include gateways with known locations in the trace reconstruction, the numbers in Table 1 may differ from those in Figure 3, which does not include known gateways in its analysis. As we increase the requisite number of interactions, we reduce the number of gateways and the amount of traces we can reconstruct, but improve the accuracy of the gateways and traces we do estimate. These results demonstrate the feasibility of such a reconstruction attack using a few well-placed gateways and a wealth of connection metadata.

4 Discussion

Industrial deployments of opportunistic networks for the long tail of mobile devices indicate that interest in this area will only continue to increase. Previously, opportunistic networks were primarily small-scale academic projects, but now commercial systems (e.g. Tile, *Find My*, Sidewalk) are widely deployed and have a large impact on personal privacy. As interest ramps up and more companies begin to support third-party backhaul, user privacy should and can be prioritized. Having demonstrated risks of these networks, we now look towards identifying current challenges and laying out a path for future work. Specifically, we explore the tradeoffs of using private information retrieval (PIR) [11] to provide *metadata privacy*, bidirectional communication to provide *accountability*, and database sharding and differential privacy [14] to provide *scalability*.

Metadata Privacy. We showed that in dense opportunistic network deployments, routing metadata alone can be used to reconstruct client mobility traces. Therefore, data-packet source identifiers and timing data should be treated as sensitive information. PIR-based schemes are a promising starting point to hide source identifiers because they allow constrained gateways to outsource compute-intensive operations that protect anonymity to the cloud. For example, *anonymous communication* systems [12, 15, 23], in which clients write data to a server, but do not reveal which client wrote a specific entry, can be used to hide source identifiers. Express [15], for example, maintains a database of mailbox rows whose contents are secret-shared between two, non-colluding servers. Riffle [24] uses a mixnet to securely shuffle client messages in a multi-server architecture to hide their source. While using anonymous communication can hide data source device identifiers, it still leaks upload timing information. An obvious approach to prevent this is hiding timing metadata by batching uploads to a cloud system at a set frequency, however, as we will discuss there are serious limitations to this approach associated with accountability and scalability.

Accountability. After decoupling device identities from any data they transmit, the next challenge lies in billing and authentication of those devices by the server based on their data patterns. Anonymous communication is one ideal solution because read-public PIR allows for authentication and the ability to track the volume of data being read from the database. This enables the network provider to charge users based on the amount of their data that is transmitted through the network. However, this still leaves open the issue of spam prevention. The threat of spam is amplified by the fact that one data transfer requires writes to many rows of the PIR database making it particularly vulnerable to DoS attacks. Importantly, if a significant amount of spam data accumulates on the server (i.e. it is not paid for by a consumer), it has already consumed a portion of the bandwidth. Therefore, we argue that, especially in an anonymous system, deny lists must reside on the gateway device as opposed to a centralized server. However, gateways are not guaranteed to always be in network coverage, making dynamically querying deny lists difficult. An additional privacy risk is that sharing deny lists based on location reveals the granular location associated with the list. To solve this issue, the server must be oblivious to which gateway is querying a particular deny list. One idea to accomplish this is to set up a bidirectional anonymous communications scheme to share location based deny lists. Deny lists could reside in queryable mailboxes to prevent the server from knowing which location list the gateway is accessing.

Scalability. There are inherent trade-offs between privacy, complexity, and performance, with stricter privacy guarantees resulting in higher computation, memory, and bandwidth cost. A system that is resistant to spam and uses anonymous communication will add computational overhead. Since guaranteeing anonymity using PIR relies on each gateway writing to multiple database rows, the number of devices that can be supported is limited. One well studied, and promising, way to address this is sharding larger databases in to smaller tables. In a PIR based system, shards could be split into smaller shards based on mailbox ID as write traffic increases. There is, however, a tradeoff to be considered with sharding: smaller tables can support fewer endpoint devices and require more server infrastructure. Furthermore if tables get too small, the locality of a shard may reveal additional location information. As noted in Section 4, batching uploads can guarantee that each gateway's timing behavior will be indistinguishable from a server standpoint. An issue with these types of approaches at scale is that cover traffic is needed to provide privacy. We propose using differential privacy based techniques to add noise to the upload time locally. By adding noise locally at the gateway, it is possible to avoid using cover traffic in exchange for a measurable privacy loss and additional latency. Due to the repetitive nature of human behavior, uses of differential privacy must take into account a degrading privacy budget with repeated uploads.

5 Conclusion

In this paper, we demonstrate that transmission metadata relayed in existing opportunistic backhaul systems leaks personal location information. Our analysis highlights a real privacy risk: with sparse gateway location knowledge, mobility traces can be recreated. Since such systems are being deployed at scale, privacy and security must be taken into consideration. We explore and discuss a path to designing privacy-aware opportunistic systems in the future.

Acknowledgements

This work was supported in part by the CONIX Research Center, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.

6

References

[1] Find my network. https://developer.apple.com/find-my/. Accessed: 09.24.2021.

[2] U.k. bars trash cans from tracking people with wi-fi. *CBS News*, 2013.

[3] Amazon sidewalk privacy and security whitepaper. https://m.media-amazon.com/images/G/01/sidewalk/final_privacy_security_whitepaper.pdf, 2021.

[4] Joshua Adkins, Branden Ghena, and Prabal Dutta. Freeloader's guide through the google galaxy. In *Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications*, pages 111–116, 2019.

[5] Joshua Adkins, Branden Ghena, Neal Jackson, Pat Pannuto, Samuel Rohrer, Bradford Campbell, and Prabal Dutta. The signpost platform for city-scale sensing. In *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 188–199. IEEE, 2018.

[6] Suzan Ali, Tousif Osman, Mohammad Mannan, and Amr Youssef. On privacy risks of public wifi captive portals. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*, pages 80–98. Springer, 2019.

[7] Thierry Antoine-Santoni, Jean Francois Santucci, Emmanuelle de Gentili, and Bernadette Costa. Using wireless sensor network for wildfire detection. a discrete event approach of environmental monitoring tool. In *2006 First International Symposium on Environment Identities and Mediterranean Area*, pages 115–120. IEEE, 2006.

[8] Apple/Google. Exposure notification - bluetooth specification. shttps://covid19-static.cdn-apple.com/applications/covid19/current/static/contact-tracing/pdf/ExposureNotification-BluetoothSpecificationv1.2.pdf?1, 2020.

[9] Andrew Blumberg and Peter Eckersley. On locational privacy, and how to avoid losing it forever. https://www.eff.org/wp/locational-privacy, 2009.

[10] J. Burgess, B. Gallagher, David D. Jensen, and B. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networks. *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, pages 1–11, 2006.

[11] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. Private information retrieval. *J. ACM*, 45:965–981, 1998.

[12] Henry Corrigan-Gibbs, Dan Boneh, and David Mazières. Riposte: An anonymous messaging system handling millions of users. In *2015 IEEE Symposium on Security and Privacy*, pages 321–338. IEEE, 2015.

[13] Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel. Unique in the crowd: The privacy bounds of human mobility. *Scientific reports*, 3(1):1–5, 2013.

[14] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9:211–407, 2014.

[15] Saba Eskandarian, Henry Corrigan-Gibbs, Matei Zaharia, and Dan Boneh. Express: Lowering the cost of metadata-hiding communication with cryptographic privacy. In *30th {USENIX} Security Symposium ({USENIX} Security 21)*, pages 1775–1792, 2021.

[16] Brian Fung. How stores use your phone's wifi to track your shopping habits. *The Washington Post*, 19, 2013.

[17] Amir Haleem, Andrew Allen, Andrew Thompson, Marc Nijdam, and Rahul Garg. Helium: A decentralized wireless network. 2018.

[18] David Hasenfratz, Olga Saukh, Silvan Sturzenegger, Lothar Thiele, et al. Participatory air pollution monitoring using smartphones. *Mobile Sensing*, 1:1–5, 2012.

[19] Wajih Ul Hassan, Saad Hussain, and Adam Bates. Analysis of privacy protections in fitness tracking social networks-or-you can run, but can you hide? In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 497–512, 2018.

[20] Alexander Heinrich, Milan Stute, Tim Kornhuber, and Matthias Hollick. Who can find my devices? security and privacy of apple's crowd-sourced bluetooth location tracking system. *arXiv preprint arXiv:2103.02282*, 2021.

[21] Jim Hildenbrand. Simplifying wireless iot gateway deployment. https://machineq.com/post/simplifying-wireless-iot-gateway-deployment, 2019.

[22] Cecilia Kang. Four attorneys general claim google secretly tracked people. https://www.nytimes.com/2022/01/24/technology/google-location-services-lawsuit.html, 2022.

[23] Albert Kwon, Henry Corrigan-Gibbs, Srinivas Devadas, and Bryan Ford. Atom: Horizontally scaling strong anonymity. In *Proceedings of the 26th Symposium on Operating Systems Principles*, pages 406–422, 2017.

[24] Albert Hyukjae Kwon, David Lazar, Srinivas Devadas, and Bryan Ford. Riffle: An efficient communication system with strong anonymity. 2015.

[25] Matthew Lentz, Viktor Erdélyi, Paarijaat Aditya, Elaine Shi, Peter Druschel, and Bobby Bhattacharjee. Sddr: Light-weight, secure mobile encounters. In *USENIX Security Symposium*, 2014.

[26] Huaxin Li, Haojin Zhu, Suguo Du, Xiaohui Liang, and Xuemin Shen. Privacy leakage of location sharing in mobile social networks: Attacks and defense. *IEEE Transactions on Dependable and Secure Computing*, 15(4):646–660, 2016.

[27] Anders Lindgren, Avri Doria, and Olov Schelén. Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE mobile computing and communications review*, 7(3):19–20, 2003.

[28] Justin Manweiler, Ryan Scudellari, and Landon P. Cox. Smile: encounter-based trust for mobile social services. In *CCS*, 2009.

[29] Jeremy Martin, Douglas Alpuche, Kristina Bodeman, Lamont Brown, Ellis Fenske, Lucas Foppe, Travis Mayberry, Erik C Rye, Brandon Sipes, and Sam Teplov. Handoff all your privacy: A review of apple's bluetooth low energy continuity protocol. *arXiv preprint arXiv:1904.10600*, 2019.

[30] Jeremy Martin, Travis Mayberry, Collin Donahue, Lucas Foppe, Lamont Brown, Chadwick Riggins, Erik C Rye, and Dane Brown. A study of mac address randomization in mobile devices and when it fails. *arXiv preprint arXiv:1703.02874*, 2017.

[31] S. Nelson, Mehedi Bakht, R. Kravets, and A. F. Harris. Encounter-based routing in dtns. *IEEE INFOCOM 2009*, pages 846–854, 2009.

[32] Nayak M Ritesh. Discontinuing support for android nearby notifications. https://android-developers.googleblog.com/2018/10/discontinuing-support-for-android.html, 2018.

[33] Semtech. Lora and lorawan. https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/, 2021.

[34] R. Shah, S. Roy, S. Jain, and W. Brunette. Data mules: modeling a three-tier architecture for sparse sensor networks. *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, 2003.*, pages 30–41, 2003.

[35] Zhihao Shen, Wan Du, Xi Zhao, and Jianhua Zou. *DMM: Fast Map Matching for Cellular Data*. Association for Computing Machinery, New York, NY, USA, 2020.

[36] Soamdeep Singha, Biswapati Jana, S. Jana, and N. Mandal. A survey to analyse routing algorithms for opportunistic network. *Procedia Computer Science*, 171:2501–2511, 2020.

[37] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *WDTN '05*, 2005.

[38] Mudhakar Srivatsa and Mike Hicks. Deanonymizing mobility traces: Using social network as a side-channel. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 628–637, 2012.

[39] Tile. How tile works. https://www.thetileapp.com/en-us/how-it-works, 2021.

[40] Lillian Tsai, Roberta De Viti, Matthew Lentz, Stefan Saroiu, Bobby Bhattacharjee, and Peter Druschel. enclosure: Group communication via encounter closures. *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, 2019.

[41] Amin Vahdat. Epidemic routing for partially-connected ad hoc networks. 2009.

[42] Mark Weiser. The computer for the 21st century. *Scientific american*, 265(3), 1991.

[43] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from gps trajectories. In *WWW '09*, 2009.