Analysis of Privacy Protections in Fitness Tracking Social Networks -or- You can run, but can you hide?

Wajih Ul Hassan, Saad Hussain, and Adam Bates, ***University Of Illinois Urbana-Champaign***

https://www.usenix.org/conference/usenixsecurity18/presentation/hassan

This paper is included in the Proceedings of the 27th USENIX Security Symposium.

August 15–17, 2018 • Baltimore, MD, USA

ISBN 978-1-931971-46-1

Open access to the Proceedings of the 27th USENIX Security Symposium is sponsored by USENIX.

Analysis of Privacy Protections in Fitness Tracking Social Networks -or- You can run, but can you hide?

*Wajih Ul Hassan** *Saad Hussain** *Adam Bates*
University of Illinois at Urbana-Champaign
{*whassan3,msh5,batesa*}*@illinois.edu*

Abstract

Mobile fitness tracking apps allow users to track their workouts and share them with friends through online social networks. Although the sharing of personal data is an inherent risk in all social networks, the dangers presented by sharing personal workouts comprised of geospatial and health data may prove especially grave. While fitness apps offer a variety of privacy features, at present it is unclear if these countermeasures are sufficient to thwart a determined attacker, nor is it clear how many of these services' users are at risk.

In this work, we perform a systematic analysis of privacy behaviors and threats in fitness tracking social networks. Collecting a month-long snapshot of public posts of a popular fitness tracking service (21 million posts, 3 million users), we observe that 16.5% of users make use of Endpoint Privacy Zones (EPZs), which conceal fitness activity near user-designated sensitive locations (e.g., home, office). We go on to develop an attack against EPZs that infers users' protected locations from the remaining available information in public posts, discovering that 95.1% of moderately active users are at risk of having their protected locations extracted by an attacker. Finally, we consider the efficacy of state-of-the-art privacy mechanisms through adapting *geo-indistinguishability* techniques as well as developing a novel EPZ fuzzing technique. The affected companies have been notified of the discovered vulnerabilities and at the time of publication have incorporated our proposed countermeasures into their production systems.

1 Introduction

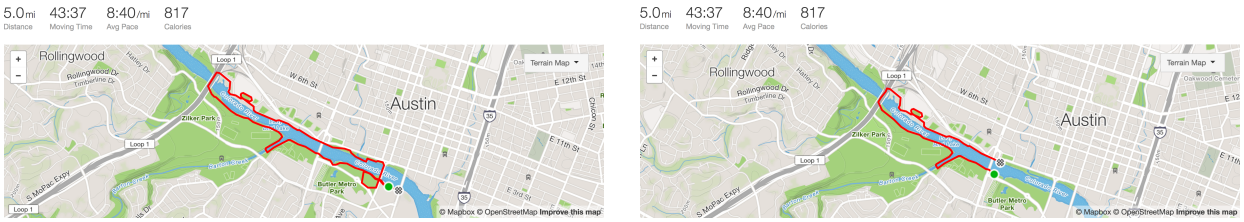
Fitness tracking applications such as Strava [23] and MapMyRide [1] are growing increasingly popular, providing users with a means of recording the routes of their cycling, running, and other activities via GPS-based tracking (i.e., *self-tracking* [44]). These apps sync to a social network that provides users with the ability to track their progress and share their fitness activities with other users. The ability to share fitness activities is an essential ingredient to the success of these services, motivating users to better themselves through shared accountability with friends and even compete with one another via leaderboards that are maintained for popular routes.

Although the sharing of personal data is an inherent risk in all social networks [42, 45, 48, 53, 56], there are unique risks associated with the data collected by fitness apps, where users share geospatial and temporal information about their daily routines, health data, and lists of valuable exercise equipment. While these services have previously been credited as a source of information for bicycle thieves (e.g., [6, 17]), the true risk of sharing this data came to light in January 2018 when Strava's global heat map was observed to reveal the precise locations of classified military bases, CIA rendition sites, and intelligence agencies [24]. Fitness activity is thus not only a matter of personal privacy, but in fact is "data that most intelligence agencies would literally kill to acquire" [46].

In response to public criticism over the global heat map incident, Strava has pointed to the availability of a variety of privacy protection mechanisms as a means for users to safeguard their accounts [50] – in addition to generic privacy settings, domain-specific mechanisms such as *Endpoint Privacy Zones* (EPZs) conceal fitness activity that occurs within a certain distance of sensitive user locations such as homes or work places [15, 16, 13]. However, at present it is unclear if such features are widely used among athletes, nor is it clear that these countermeasures are adequate to prevent attackers from discovering the private locations of users.

In this work, we perform a systematic analysis of privacy threats in fitness tracking social networks. We begin by surveying the fitness app market to identify classes of privacy mechanisms. Using these insights, we then formalize an attack against the Endpoint Privacy Zones fea-

*Joint first authors.



(a) Without Privacy Zone

(b) With Privacy Zone

Figure 1: Summary of a Strava running activity that occurred in Austin, Texas during USENIX Security 2016. Figure 1a displays the full exercise route of the athlete. Figure 1b shows the activity after an Endpoint Privacy Zone (EPZ) was retroactively added, obscuring the beginning and end parts of the route that fell within $\frac{1}{8}$ miles of the Hyatt Regency Austin hotel.

ture. To characterize the privacy habits of users, we collect a month-long activity dataset of public posts from Strava, an exemplar fitness tracking service. We next use this dataset to evaluate our EPZ attack, discovering that 95.1% of regular Strava users are at risk of having their homes and other sensitive locations exposed. We demonstrate the generality of this result by replicating our attack against data collected from two other popular fitness apps, Garmin Connect and Map My Tracks.

These findings demonstrate privacy risks in the state-of-the-practice for fitness apps, but do not speak to the state-of-the-art of location privacy research. In a final series of experiments, we leverage our Strava dataset to test the effectiveness of privacy enhancements that have been proposed in the literature [26, 27]. We first evaluate the EPZ radius obfuscation proposed by [27]. Next, we adapt spatial cloaking techniques [41] for use in fitness tracking services in order to provide geo-indistinguishability [26] within the radius of the EPZ. Lastly, we use insights from our attack formalization to develop a new privacy enhancement that randomizes the boundary of the EPZ in order to conceal protected locations. While user privacy can be improved by these techniques, our results point to an intrinsic tension that exists within applications seeking to share route information and simultaneously conceal sensitive end points.

Our contributions can be summarized as follows:

- *Demonstrate Privacy Leakage in Fitness Apps.* We formalize and demonstrate a practical attack on the EPZ privacy protection mechanism. We test our attack against real-world EPZ-enabled activities to determine that 84% of users making use of EPZs unwittingly reveal their sensitive locations in public activity posts. When considering only moderate and highly active users, the detection rate rises to 95.1%.

- *Characterize Privacy Behaviors of Fitness App Users.* We collect and analyze 21 million activities representing a month of Strava usage. We characterize demographic information for users and identify a significant demand for privacy protections by 16.5%, motivating the need for further study in this area.

- *Develop Privacy Extensions.* Leveraging our dataset of public activity posts, we evaluate the effectiveness of state-of-the-art privacy enhancements (e.g., geo-indistinguishability [26]) for solving problems in fitness tracking services, and develop novel protections based on insights gained from this study.

- *Vulnerability Disclosure.* We have disclosed these results to the affected fitness tracking services (Strava, Garmin Connect, and Map My Tracks). All companies have acknowledged the vulnerability and have incorporated one or more of our proposed countermeasures into their production systems.[1]

2 Fitness Tracking Social Networks

Popularized by services such as Strava [23], fitness tracking apps provide users the ability to track their outdoor fitness activities (e.g., running) and share those activities with friends as well as other users around the world. Leveraging common sensors in mobile devices, these services track users' movements alongside other metrics, such as the altitude of the terrain they are traversing. After completing a fitness activity, users receive a detailed breakdown of their activities featuring statistics such as distance traveled. If the user pairs a fitness monitor (e.g., Fitbit [3]) to the service, the activity can also be associated with additional health metrics including heart rate. Beyond publishing activities to user profiles, fitness tracking services also offer the ability for users to create and share recommended routes (segments). Each segment is associated with a leaderboard that records the speed with which each user completed it. Most fitness tracking services also contain a social network platform through which users can follow each other [12, 18, 23].

[1] A summary of the disclosure process as well a statement on the ethical considerations of this work can be found in Section 9.

App Name	# D/Ls	Private Profiles	Private Activities	Block Users	EPZs	Radius Sizes [min,max], inc
Strava [23]	10M	✓	✓	✓	✓	[201,1005], 201
Garmin [12]	10M	✓	✓	✗	✓	[100,1000], 100
Runtastic [22]	10M	✓	✗	✓	✗	-
RunKeeper [21]	10M	✓	✓	✗	✗	-
Endomondo [20]	10M	✗	✓	✗	✗	-
MapMyRun [1]	5M	✓	✓	✗	✗	-
Nike+ [7]	5M	✓	✓	✗	✗	-
Map My Tracks [18]	1M	✗	✓	✗	✓	[500,1500], 500

Table 1: Summary of privacy features offered across different popular fitness tracking services. #D/Ls: downloads (in millions) on Android Play store. EPZ radius given in meters.

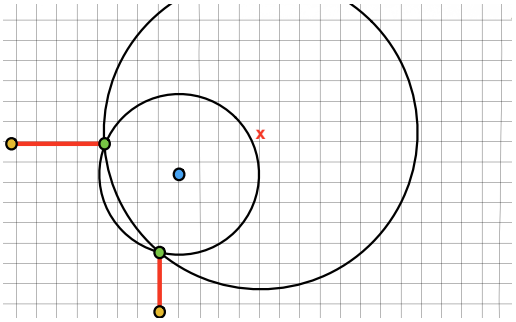
Followers are granted additional access to user information that may not be publicly available, such as the list of equipment that the user owns.

As is evident from the features described above, fitness tracking services share a variety of highly sensitive user information, including spatial and temporal whereabouts, health data, and a list of valuable equipment that is likely to be found in those locations. Recognizing the sensitivity of this information, these services offer a variety of privacy mechanisms to protect their users. We conducted a survey of privacy mechanisms across 8 popular fitness networks, and present a taxonomy of these features in Table 1. Popular mechanisms include:

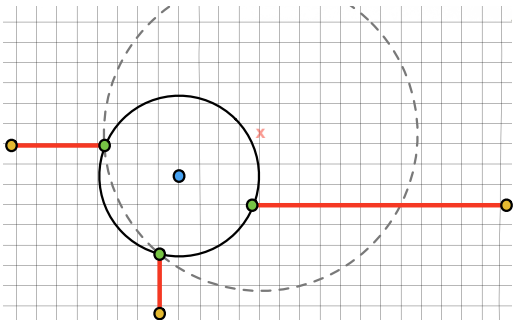
F1 Private Profiles/Activities: As is common across many social networks, users have the ability to make their posts or profiles private. Depending on the service, users can elect to make all activities private or do so on a case-by-case basis. However, hidden activities are not counted towards challenges or segment leaderboards, incentivizing users to make their activities public. Of the surveyed services, only Garmin Connect enables private activities by default.

F2 Block Users: Like other social networks, users have the ability to block other users, removing them from their follower's list, and preventing them from viewing their activities or contacting them. However, as posts are public by default on many services, the ability to block a user offers limited utility.

F3 Endpoint Privacy Zone: Since users will often start their activities at sensitive locations, several services allow users the option to obfuscate routes within a certain distance of a specified location. In this paper, we refer to this general mechanism as an *Endpoint Privacy Zone (EPZ)* [15]. If an activity starts or ends within an EPZ, the service will hide the portion of the user's route within the EPZ region from being viewed by other users. We provide a formal definition of an EPZ in Section 3. An example is shown in Figure 1; after enabling an EPZ, the full route (Fig. 1a) is truncated such that segments of the route are not visible within a certain radius of a sensitive location (Fig. 1b).[2] Unfortunately, there are also disincentives to leveraging the privacy zones. For example, Strava and Garmin Connect users will not appear on leaderboards for routes that are affected by their privacy zone.



(a) With fewer activities, there are multiple possible EPZs.



(b) As activities increase, possible EPZs are eliminated.

Figure 2: Simplified activity examples that demonstrate the intuition behind our EPZ identification approach. Red lines represent activity routes, while circles represent possible EPZs. In Fig. 2a, given the available routes there are multiple possible EPZs of different radii, only one of which is correct. In Fig. 2b, an additional activity reduces the space of possible EPZs to one.

F4 EPZ Radius Size: All three services (Strava, Garmin Connect, Map My Tracks) that provide an EPZ feature, allow users the option of selecting a circular obfuscation region from a fixed set of radius size values. Different services provide different minimum and maximum radius sizes with fixed increments to increase and decrease the size of EPZ radius. For example, Garmin Connect allows users to select a minimum and a maximum radius of 100 and 1000 meters with 100 meters increments.

[2] These images are being used with the permission of the athlete and do not leak any personally identifiable information as the pictured activity took place on site at a conference.

3 You can run, but can you hide?

In this section, we set out to determine whether or not fitness tracking services' users' trust in the EPZ mechanism is misplaced. To do so, we present an efficient attack methodology for identifying EPZs. As discussed in Section 2, EPZs place a hidden circle around the user's private location in order to prevent route data within a given radius of that location from appearing on activity webpages. The hidden part of the route is only visible to the owner of the activity. Moreover, the number of allowed EPZ radius sizes are fixed based on the fitness tracking service. For example, Strava provides a fixed set of EPZ radii of $\frac{1}{8}$, $\frac{1}{4}$, $\frac{3}{8}$, $\frac{1}{2}$, or $\frac{5}{8}$ of a mile.

It may be intuitive to the reader that, given a finite set of possible circle radii and a handful of points that intersect the circle, the center of the circle (i.e., a user's protected location) is at risk of being inferred. Figure 2 demonstrates this intuition for EPZs. When only one route intersection point is known, there is a large space of possible EPZ locations; however, given two intersection points, the number of possible EPZs is dramatically reduced, with the only remaining uncertainty being the radius of the circle (Figure 2a). Given three distinct intersection points (Figure 2b), it should be possible to reliably recover the EPZ radius and center.

In spite of this intuition, it is not necessarily the case that EPZs are ineffective in practice; a variety of factors may frustrate the act of EPZ identification. First, *services that offer EPZ mechanisms do not indicate to users when an EPZ is active on a route.* Instead, as shown in Figure 1, the route is redrawn as if the activity started and finished outside of the invisible EPZ. Even if an activity is known to intersect an EPZ, it is not obvious which side of the route (beginning or end) the EPZ intersects. Activity endpoints that intersect an EPZ are therefore indistinguishable from endpoints that do not, creating significant noise and uncertainty when attempting to infer a protected location. Moreover, the GPS sampling fidelity provided by fitness tracking devices and services may be such that the exact point where a route intersects an EPZ may be irrecoverable. Alternately, it may also be that EPZs are recoverable in only highly favorable conditions, making the identification of fitness tracking service users at scale impractical.

3.1 Threat Model

We consider an adversary that wishes to surreptitiously identify the protected home or work locations of a target user on a fitness tracking service. Through the use of a dummy account, the adversary learns how the fitness tracking service protects private locations, as described in Section 2. However, the attacker is unaware of the target user's protected location, and moreover is uncertain if the target has even registered a protected location. To avoid arousing suspicion, the attacker may surveil the target user in any number of ways – by following the user's profile from their own account, or querying the target user's data via a service API. Regardless of the means, the singular goal of the adversary is to determine the existence of an EPZ and recover the protected address using only fitness activities posted to the users' account.

3.2 Breaking Endpoint Privacy Zones

Problem Formulation. We formulate our problem as the *EPZ Circle Search Problem* in the Cartesian plane. We convert GPS coordinates of the activities to Earth-Centered Earth-Fixed (ECEF) coordinates in the Cartesian plane. The details of conversion can be found in [57]. This is justified by the fact that both services and protocols such as GPS cannot provide arbitrary accuracy. Moreover, this makes the attack algorithm calculations easier without loss of important information. We first proceed to give a formal definition of EPZ and use this definition for remainder of section.

Definition 1. ***Endpoint Privacy Zone.*** *Let point $p_s = (x_s, y_s)$ be a sensitive location in the Cartesian plane, and a be an activity route of n points $< p_1, \ldots, p_n >$. $EPZ_{p_s,r}$ is a circle with center p_s and radius r that is applied to activity a if p_1 or p_n are within distance r of p_s. If this is the case, all points p_i in a that are within distance r of p_s are removed from a.*

With this in mind, the definition of the EPZ Circle Search Problem is as follows:

Definition 2. ***EPZ Circle Search Problem.*** *Let $EPZ_{p_s,r}$ be an active EPZ where r is in the set R_S provided by service S, and let A_u be the set of activity routes for user u of the form $< p_1, \ldots, p_n >$. In the EPZ search problem, the goal is to guess $(p_g, r_g \in R_S)$ such that EPZ_{p_g,r_g} best fits endpoints p_1 and p_n for all activities in A_u.*

In order to identify a suitable algorithm for EPZ search problem, we first looked into *circle fit* algorithms. Circle fit algorithms take sets of Cartesian coordinates and try to fit a circle that passes through those points. The most studied circle fit algorithm is *Least Squares Fit (LSF)* [40] of circle. This method is based on minimizing the mean square distance from the circle to the data points. Given n points (x_i, y_i), $1 \leq i \leq n$, the objective function is defined by

$$F = \sum_{i=1}^{n} d_i^2 \tag{1}$$

where d_i is the Euclidean (geometric) distance from the point (x_i, y_i) to the circle. If the circle satisfies equation

$$(x-a)^2+(y-b)^2=r^2 \quad (2)$$

where (a, b) is its center and r its radius, then

$$d_i=\sqrt{(x_i-a)^2+(y_i-b)^2}-r \quad (3)$$

Limitations of LSF. The minimization of equation 1 is a nonlinear problem that has no closed form solution. There is no direct algorithm for computing the minimum of F, all known algorithms are iterative and costly by nature [32]. Moreover, the LSF algorithm also suffers from several limitations when applied to EPZ Circle Search Problem. The first limitation is that the adversary is not sure which points in an activity intersect the EPZ. There can be up to 4 endpoints in a modified route, but at most two of these points intersect the EPZ. Feeding one of the non-intersecting points into LSF will lead to an inaccurate result. Therefore, the adversary must run the LSF algorithm with all possible combinations of endpoints and then pick the result that minimizes F. However, we discovered through experimentation that the LSF algorithm is prohibitively slow for large sets of activities. The third limitation is that LSF considers circles of all possible radii. However, in the case of fitness tracking services context, the algorithm need only consider the small finite set of radii R_S.

In order to overcome above limitations, we devised a simpler and more efficient algorithm that fits our needs. We will first give a strawman algorithm to search EPZ then we will refine this algorithm in various steps.

ALGORITHM STRAWMAN. Given a set of activities A_u and possible radii R_S, iterate through pairs of activities and perform pairwise inspection of each possible combination of endpoints. For each pair of endpoints $(x_1,y_1),(x_2,y_2)$, solve the simultaneous equations:

$$(x_c-x_1)^2+(y_c-y_1)^2=r^2 \quad (4)$$

$$(x_c-x_2)^2+(y_c-y_2)^2=r^2 \quad (5)$$

where r is one of the radius from R_S and (x_c,y_c) is the center of a possible EPZ. Store each solution for the simultaneous equations as a candidate EPZs in set *SS*. When finished, return a randomly selected item in *SS* as a guess for the protected location.

Refinement #1 (Confidence Score & Threshold): The above algorithm is not deterministic – multiple EPZs are predicted by the algorithm, but only one is the correct one for the given user u. Pruning these possibilities requires the introduction of a metric to indicate that one candidate EPZ is more likely to be correct than the others. We observe that the correct EPZ prediction will occur most often; this is because all endpoint pairs that intersect the EPZ will produce the same result, whereas endpoint pairs that do not intersect the EPZ will produce different results each time. Therefore, we introduce a consensus procedure to select our prediction from the set of candidate EPZs. A *confidence score* is assigned to each EPZ, where the value of this metric is the number of activity start/end points that independently agree on the location of the EPZ. To prevent our algorithm from issuing a bad prediction when insufficient information (i.e., activities) is available, we also introduce a *confidence threshold* τ_c. τ_c represents the minimum confidence score needed to qualify as an EPZ prediction. If a candidate EPZ is *less* than the confidence threshold, then it is removed from consideration. The final prediction of the algorithm, if any, is the candidate EPZ with the highest confidence score exceeding t_c, as shown in line 28 of Algorithm 1.

Refinement #2 (Distance Similarity Threshold): Due to sampling noise and imprecision in the GPS coordinates made available by fitness tracking devices/services, it may be that activity endpoints do not lie exactly on the EPZ circle. As a result, our algorithm

Algorithm 1: EPZ Search Algorithm

```
   Inputs : A_u, τ_d, τ_c, τ_i, R_S
   Output: KeyValueStore of EPZ, confidence level
1  PossibleEPZs ← KeyValueStore()
2  foreach (A_1, A_2) ∈ A_u do
       /* 6 possible point pairs are generated.                    */
3      PointPairs ← Pairs of start and end points from A_1 and A_2
4      foreach PointPair ∈ PointPairs do
           /* For each possible EPZ radius.                        */
5          foreach r ∈ R_S do
6              SS ← Solve simultaneous eq. for r, PointPair
7          end
8      end
9      foreach EPZ ∈ SS do
10         PossibleEPZs[EPZ] ← 1
11     end
12 end
13 foreach EPZ ∈ PossibleEPZs do
14     foreach (A) ∈ A_u do
           /* Haversine formula calc. dist. between coords.        */
           /* Refinement #3                                        */
15         if EPZ.R − Haversine(EPZ,A) > τ_i then
16             Delete(PossibleEPZs[EPZ_2])
17     end
18 end
19 foreach EPZ_1 ∈ PossibleEPZs do
20     foreach EPZ_2 ∈ PossibleEPZs do
21         if EPZ_1 ≠ EPZ_2 then
               /* Refinement #2                                    */
22             if Haversine(EPZ_1,EPZ_2) < τ_d then
23                 PossibleEPZs[EPZ_1] += PossibleEPZs[EPZ_2]
24                 Delete(PossibleEPZs[EPZ_2])
25     end
26 end
27 foreach key, value ∈ PossibleEPZs do
       /* Refinement #1                                            */
28     if value < τ_c then
29         Delete key from PossibleEPZs
30 end
31 return PossibleEPZs
```

will predict slightly different p_g values for different endpoints pairs, even when considering endpoints that truly intersect the EPZ. Our algorithm will not be able to accumulate confidence in a given prediction unless we can account for this noise. Therefore, we introduce a *distance similarity threshold* τ_d. When comparing two candidate EPZs to one another, the refined algorithm considers two circles as same if the distance between the centers is less than or equal to this threshold. τ_d is used in the Algorithm 1 from line 19 to line 26.

Refinement #3 (Activity Intersection Threshold): To reduce the space of candidate EPZs, we can leverage the knowledge that no endpoint from any activity in the set A_u should fall within the candidate EPZ's circle, as this necessarily implies that an EPZ was not active in that area for user u. However, we must also account for measurement error when performing this test – due to noise in GPS sampling, there is a chance that an activity passing nearby the area of the candidate EPZ could produce endpoints that appear to lie within the circle. This would result in ruling out a candidate EPZ that may in fact be the true EPZ. To mitigate this problem, we introduce an *activity intersection threshold* τ_i. Our refined algorithm does consider an endpoint to intersect a candidate EPZ unless it falls more than τ_i within the EPZ circles, as shown in the Algorithm 1 from line 13 to line 18.

ALGORITHM REFINED. Extending our original strawman algorithm, our final refined algorithm is shown in Algorithm 1. Given as input a set of activities for a single user A_u, distance similarity threshold τ_d, activity intersection threshold τ_i, confidence threshold τ_c, and set of EPZ radii R_S, the algorithm returns all the candidate EPZs with their confidence value, with the highest confidence point p_g representing a prediction for u's protected location. Note that value of thresholds depend on the fitness tracking service and require training runs to parameterize. We will describe our procedure for finding these threshold values in Section 5.

4 Data Collection[3]

To evaluate the plausibility of the above EPZ attack algorithm, we require a large corpus of naturalistic usage data for a fitness tracking app. Strava is one of the most popular fitness tracking apps, with over a million active monthly users [2] and over a billion total activities recorded so far. We thus select it as an exemplar fitness tracking app.[4] In this section, we describe our methodology for collecting usage information from public Strava posts. In characterizing the resulting dataset, we also provide useful insights as to the privacy habits of the athletes on fitness tracking apps.

4.1 Methodology

We begin by collecting a large sample of public posts to the Strava using a `cURL`-based URL scraping script. Because Strava assigns sequential identifiers to activities as they are posted, our scraper was able to traverse posts to the network in (roughly) chronological order. It was also able to obtain data resources for each post in JSON-encoded format using an HTTP REST API. Our scraper did not collect data from private activities, only the information available in public posts. In fact, it was not necessary to be logged into Strava in order to access the sites visited by our scraper. These features have previously been used by other members of the Strava community in order to measure various aspects of the service [8, 9, 10].

The scraper takes as input a start and an end activity ID, then iterates across the continuous sequence of activity IDs. For each ID, the crawler first visits the `strava.com/activities/ID` page to extract the activity's start date and time, Athlete ID, total distance, total duration, reported athlete gender, and the type of the activity. It then uses the `strava.com/stream/ID` API to extract GPS samples for the activity route, as well as the total distance traveled at each GPS sample. The scraper uses the first GPS coordinate in the route to obtain the country of the activity. Using an additional API that facilitates interoperability between Strava and other social networks, the scraper recovers the time the activity was posted, then subtracts the length of the activity to approximate the start time. Through experimentation, we discovered that when an activity is associated with an EPZ, *there is a discrepancy between the advertised distance on the activity page and the final distance traveled according to the GPS samples*; the crawler check-marks the activity as EPZ-enabled if this discrepancy is found.

4.2 Data Corpus

Using the above methodology, we collected a month worth of Strava activities beginning on May 1, 2016. The activity IDs associated with May 1 and May 31 were identified by conducting a binary search of the activity space and verified through manual inspection. However, we note that activity IDs are assigned in *roughly* chronological order; we observed activities that appeared months to years out of sequence. We attribute this behavior to devices that had intermittent network connectivity and to users that deliberately set their device to the incorrect date. It is therefore likely that our dataset omits a small percentage of activities that occurred in May 2016. Scraped activities that fell outside of May 2016 were dis-

[3] This section describes a methodology that is no longer feasible on Strava following changes made in response to our disclosure.

[4] Although our approach is primarily evaluated on Strava, note that in § 7 we demonstrate the generality of the attack using other services.

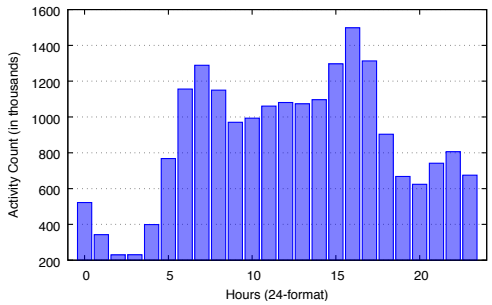


Figure 3: Distribution of Activities by time of the day.

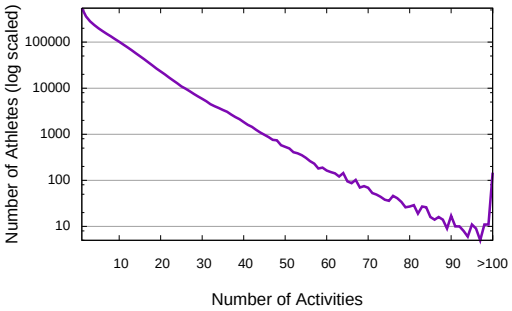


Figure 4: Distribution of Athletes by activities recorded.

carded from the dataset. Running our scraper across 15 CPU threads, the dataset took 14 days to collect.

Initially, the dataset contained over 23,925,305 activities. Three types of activities were discarded: *1) Private activities* for which we did not retrieve any usage information, *2) Activities with 0.0 distance* that did not have any route information, and *3) Activities with type other than Walk, Ride, and Run* . We observed 8 different activity types (Ride, Run, Walk, Hike, Virtualride, Swim, Workout, and others) in our dataset, with Ride, Run, and Walk comprised the 94% of total activities. Other activity types (e.g., workouts) were excluded because they were unlikely to be actual GPS routes or protected locations, while others (e.g., Virtual-ride) likely reported false GPS routes. The remaining dataset contained 20,892,606 activities from 2,960,541 athletes.

We observed a total of 2,360,466 public activities that were associated with an EPZ; as a point of comparison, this is more than twice the number of (excluded) private activities (1,080,484), underscoring the popularity of the EPZ feature. The use of EPZs is spread out across a large number of users, with 432,022 athletes being associated with at least one EPZ activity and 346,433 being associated with more than one EPZ activity. Total activities by male-identifying athletes are 16,703,160 and female-identifying are 3,227,255, while 962,191 activities report no gender identity. A diurnal pattern is observable in the distribution of activities by time of day, as shown in Figure 3. 545,997 users are not regularly active in our dataset, logging only one activity; however, as shown

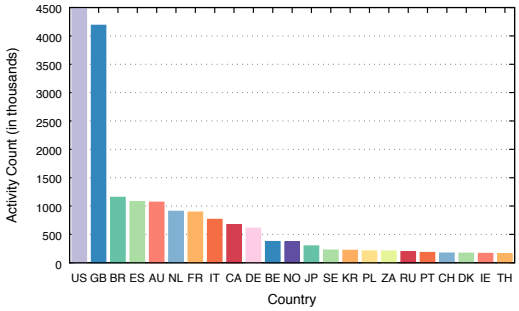


Figure 5: Most popular countries in our dataset.

in Figure 4, the dataset reflects a healthy variety of usage levels, with many athletes logging over 100 activities during the month. We also note the the diverse demographic makeup of our dataset. Figure 5 shows the international popularity of Strava. While the United States (US) and Great Britain (GB) are the most active countries by a significant margin, 21 other countries contain at least 150,000 activities, with 241 countries appearing in the dataset overall.[5]

5 Evaluation[6]

We now leverage our activity dataset comprised of Strava public posts to perform a large-scale privacy analysis of EPZ mechanism. To establish ground truth with which to measure the accuracy of our EPZ identification algorithm, we first create a synthetic set of EPZ-enabled activities using unprotected routes for which the true endpoints are known. After validating our approach, we then quantify the real-world severity of this danger by running our algorithm against legitimate EPZ-enabled activities. We discover that the EPZ mechanism in fact leaks significant information about users' sensitive locations to the point that they can be reliably inferred using only a handful of observations (i.e., activities).

5.1 Validation

In order to verify that our algorithm works as intended, we require a ground truth that will enable us to issue predictions over EPZs with known centers. To do so, we make use of the 18,532,140 unprotected activities generated by 2,528,519 athletes in our Strava dataset. For each athlete, we search across their activities for endpoints that fall within 50 meters of one another; this distance approximates size of a suburban house plot. We then designate the centroid of these points as a protected

[5] While we took every effort to remove virtual activities from our dataset, we do not rule out the possibility that some activities were generated by exercise equipment training routines.

[6] This section describes the results based on Strava's previous EPZ mechanism, which was replaced following our vulnerability disclosure.

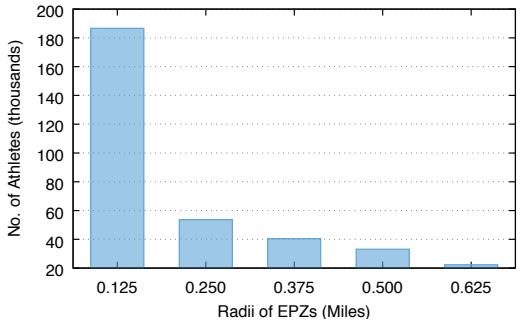


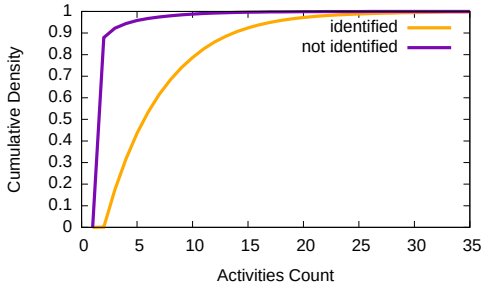
Figure 6: Distribution of identified EPZs by radius. This finding suggests that the smallest privacy zone is significantly more popular than larger privacy zones.

location, synthesize an EPZ with a radius of 0.25 miles over the centroid, and update the GPS data by removing all points that fall within the synthetic EPZ. Finally, our identification algorithm attempts to independently predict the (known) center of each (synthesized) EPZ.

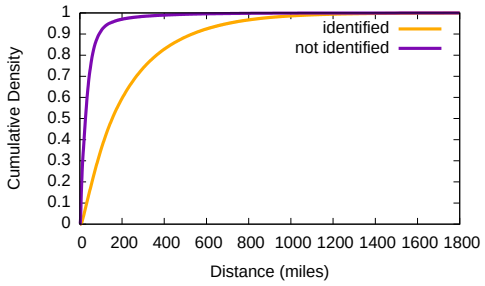
As discussed in Section 3, our algorithm is parameterized by three thresholds: t_d, t_c, and t_i. To determine effective values for these parameters, we withheld from the above synthetic data a set of 10,000 athletes. We determined that an appropriate value for the distance threshold t_d was 0.05 meters and t_i was 0.1 meters. We set our confidence threshold t_c to 3, because our predictions were never conclusive using just two activities, as discussed below. We note that these values need to be adjusted for different services, or as Strava modifies the sampling/precision of its GPS coordinates[7]. Using these parameters, we were able to identify 96.6% athletes out of 2,518,519. As noted previously, our identification algorithm is not deterministic; however, by selecting the highest confident candidate EPZ, we were able to correctly predict 96.6% of EPZs in the synthesized set.

Failure Conditions. For 3.4% of athletes, we were unable to identify an EPZ. The reason for this is almost entirely due to a lack of available observations. If only two activities were available for a given athlete, it was common that only two points would intersect the EPZ. With only two intersection points, five candidate EPZ of equal likelihood are discovered, one for each of the possible radii. This motivates our decision to set t_c to 3, as it removes a failure condition that would lead to a high false positive rate in the subsequent tests. Only in rare instances were more than two intersections obtained from just two activities.

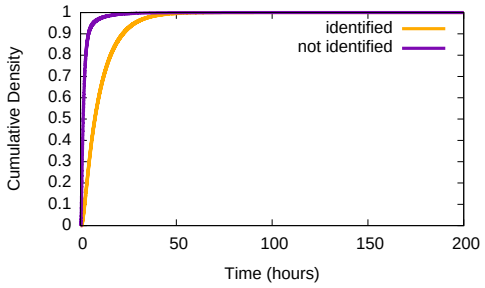
[7]Between our preliminary experiments and data collection, Strava increased the granularity of their sampling rate by a factor of 5.



(a) Identification rate by Activity Count.



(b) Identification rate by Total Distance.



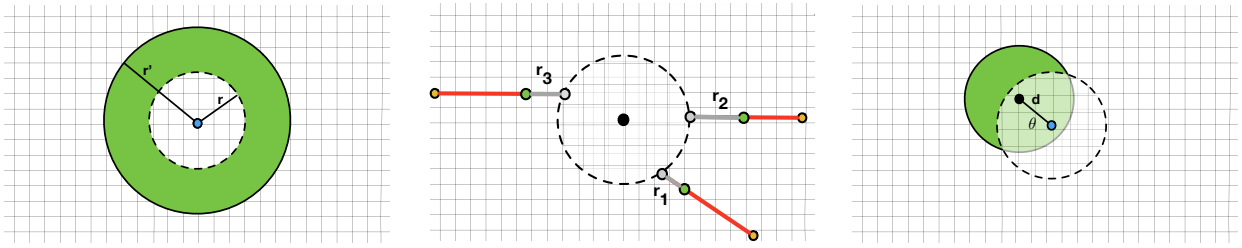
(c) Identification rate by Total Duration.

Figure 7: CDFs for identified versus unidentified locations across various metrics. Activity count is the greatest predictor of successful identification, suggesting that our technique would be more successful over a longer period of observation.

5.2 Results for EPZ Identification

Having validated the effectiveness of our technique against a synthesized dataset, we now turn our attention to identifying actual protected locations of actual Strava athletes. We ran our algorithm, as parameterized above, against our dataset of 2,360,466 EPZ-enabled activities generated by 432,022 athletes. Using our technique, we were able to identify 84% of all users protected locations with more than one EPZ-enabled activity. Under favorable conditions in which a user records at least 3 EPZ-enabled activities, our accuracy increases to 95.1%.

Figure 6 summarizes the protected locations identified by EPZ radius size. As we will demonstrate in Section 6, the effectiveness of our algorithm degrades against large EPZ radii, due solely to their propensity to obscure entire activities; in fact, for EPZ radii of 0.625 miles, we see the accuracy of our approach falls to 44% against



(a) Modify Radius **(b)** Fuzz EPZ Intersection Points **(c)** Spatial Cloaking

Figure 8: Obfuscation techniques for EPZs. The original EPZ circle is shown in white, while the enhanced EPZ circle is shown in green. In Figure 8b, the circle is unmodified but each activity route truncated by a random number of coordinates.

synthetic data. However, this decrease in efficacy alone does not account for the large difference in frequency of EPZ size. For example, if each radius were equally popular, we would have expected to identify 80,000 athletes with the 0.625 mile radius. As a result, this figure most likely reflects the distribution of EPZ radii popularity. We therefore infer that the smallest EPZ is several times more popular than any other EPZ size, and that the popularity of EPZs are inversely correlated to their radii.

We also wished to characterize the circumstances under which our technique succeeded and failed. Figure 7 shows the cumulative density functions (CDFs) of identified locations and unidentified locations across several different potentially influential metrics: the activities count for the athlete (Fig. 7a), the total distance traveled by the athlete (Fig. 7b), and the total duration of athlete activity (Fig. 7c). The greatest predictor of whether or not a protected location is leaked is the total number of activities observed. Locations that were not identified had an average of 4.6 activities, whereas locations that were identified had an average of 6.2 activities. Recall our dataset is comprised of a single month of Strava activity; this finding indicates that, over a prolonged window, the number of leaked locations is likely to be much larger than 95.1% amongst regular users of Strava.

Failure Condition. For 16% of the 432,022 total athletes that logged an EPZ-enabled activity, we were unable to detect the protected location. The reason for this is, like in our validation study, there were a number of athletes with too few activities to exceed the t_c confidence threshold. Out of the total number of athletes, we found that 11% had recorded 1 activity and out of this set, zero protected locations were identified. To demonstrate, we filtered low-activity athlete accounts and considered only the remaining 283,920 athletes. Our algorithm identified 95.1% of the protected locations for these moderately active users (3+ EPZ-enabled activities). The remaining 4.9% are accounted for by athletes that logged a single activity for multiple distinct EPZs that did not intersect. For example, one athlete recorded an EPZ-enabled activity in two different cities. *These findings indicate that the EPZ mechanism is ineffective even for moderately active users of fitness tracking services.*

6 Countermeasures

While the EPZ mechanism is widely used by fitness tracking services, it lags behind the state-of-the-art in location privacy research. In this section, we address this gap in the literature by testing state-of-the-art privacy mechanisms against our Strava dataset, as well as proposing our own defense that fuzzes the boundaries of EPZs in order to frustrate our attack.

6.1 Obfuscation techniques

Location obfuscation techniques are complementary to anonymity; rather than hiding user identities, location obfuscation techniques assume that user identities exist but add uncertainty and randomness in collected locations to decrease accuracy. Figure 8 shows the intuition of the three approaches that we consider.

1. **Modify Radius Size.** Ardagna *et al*. propose location privacy for fitness tracking domains [27] by applying a modification to the EPZ radius to enlarge the privacy zone, as shown in the Figure 8a. Here, r is the original radius of privacy zone and r' is the enlarged radius. This technique predicts that the protected location will be harder to guess if the last visible point in the activity is further away from location.

2. **Fuzz EPZ Intersection Points**: The surveyed EPZ implementations provide a GPS coordinate in the activity route that falls very close to the boundary of the privacy zone. We reason that perturbing the boundary of the EPZ will significantly increase the difficulty of attack. We therefore present a fuzzing method that, for each posted activity, randomly removes a small number of GPS coordinates beyond the true boundary of the EPZ. We predict that a small amount of

noise (e.g., a few meters) injected in this fashion will dramatically change the location of the attacker's prediction (e.g., a few blocks).

3. **Spatial Cloaking** Another technique of location obfuscation is spatial cloaking [41]. We adapt spatial cloaking the in context of fitness tracking services. We shift the center of EPZ, concealing the protected location at an unknown point within the privacy zone. This obfuscation is shown in Figure 8c, where d is the size of the shift and θ is the direction (angle) in which center moves. Note that while shifting center, the d needs to be always less than the radius of previous privacy zone circle otherwise user sensitive location information will not be obfuscated. We pick d using random value generated from Laplacian distribution to achieve ε-geo-indistinguishability where ε is level of privacy [26].[8]

6.2 Data Synthesis

To test the above privacy extensions, we generated obfuscated privacy zone records using our Strava dataset using 18,532,140 unprotected (not-EPZ enabled) activities. The reason for using unprotected activities is that they provided known locations to use as ground truths, and also because some countermeasures may actually reveal parts of the true route that were concealed by Strava's EPZ implementation. We generated a synthetic dataset using the same technique described in Section 5.1. For each user, we searched their activities for route endpoints that fell within 50 meters of one another. We took the centroid of these points and designated it as a synthetic protected location. By considering only those activities associated with one of these protected locations, our subsequent analysis was based off 1,593,364 users and associated activities. Finally, we applied a privacy-enhanced EPZ to each protected location as described below.

6.3 Countermeasure Implementations

Modify Radius. For each user, we apply each of the 5 EPZ radii permitted by Strava, which enables us to see the affect of radius size on accuracy.

Fuzz EPZ Intersection Points. After removing points from each route that fall within the EPZ, we continue to remove points up to a random distance r_i past the intersection (see Figure 8b) where $0 < r_i < F$. We initially set F to 80 meters, a value intended to approximate the size of a city block.

[8] This technique provides similar operational semantics to Ardagna et al.'s "shift center" obfuscation [27].

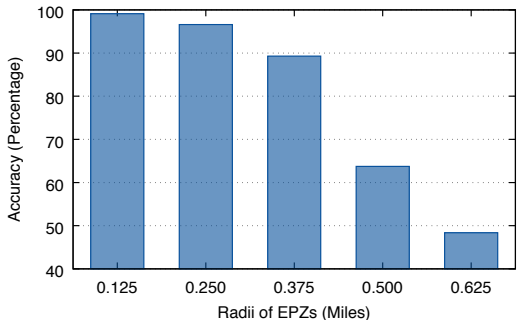


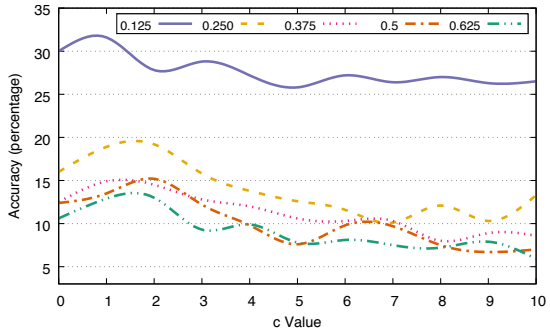
Figure 9: Efficacy of *Modify Radius* defense – while larger EPZ radii seem to reduce attack accuracy, the larger radii are actually just enveloping entire activities.

Spatial Cloaking. For each user, we choose a random radius r' from the set of permissible EPZ radii on Strava, a random angle θ ranged from 0 to 355 by factors of 5, and a random value d where $0 < d < r'$. We then shifted the center of the EPZ by distance d in the direction of θ. This ensured that the EPZ still covered the user's protected location, but that location was at a random point within the EPZ instead of the center. d was generated using a Planar Laplacian mechanism [26] to achieve ε-geo-indistinguishability. This function takes ε which was set to 1 and r which was set to r'. Finally, we truncated all user activities such that no GPS coordinate fell within the enhanced EPZ.

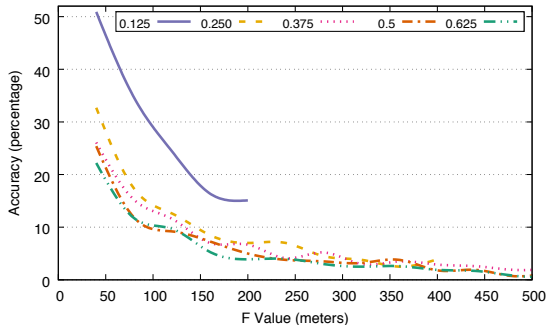
6.4 Countermeasure Evaluation

Modify Radius. Against this obfuscation, we deployed our original EPZ identification attack as described in in Section 3. The results are shown in Figure 9; while our accuracy is at 99% against 0.125 mile EPZs. our effectiveness plummets to 46% against 0.625 mile EPZs. This finding would seem to suggest that a viable and immediately applicable countermeasure against EPZ identification is simply to use one of the large radius options that are already made available by Strava. Unfortunately, upon further analysis we discovered that this was not the case. This drop in accuracy is not a result of the increased distance between endpoints and the protected location, but simply that the larger radii will often completely envelope a posted activity. In other words, the loss of accuracy can be accounted for by a decrease in observable routes (and their endpoints). At 0.625 miles, the majority of the activities in our dataset become invisible, dealing a major blow to the utility of the fitness tracking service.

Fuzz EPZ Intersection Points. Against this obfuscation, we considered that an attacker may try to account for the added noise by modifying the distance similarity threshold τ_d used in the EPZ identification algorithm. We considered a simple extension where τ_d incorporated



(a) Fixed fuzz value $F = 80$, variable constant factor c



(b) Fixed constant factor $c = 1$, variable fuzz value F

Figure 10: Efficacy of *Fuzz EPZ Intersection Points* defense. Each line charts performance using a different EPZ radii.

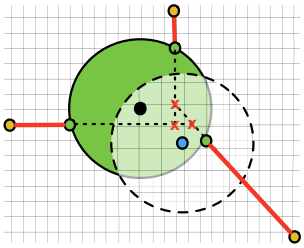


Figure 11: Activity example that demonstrates an attack against the *Spatial Cloaking* defense. If routes are moving in the direction of the protected location when they cross the EPZ, linear interpolation of the routes will yield an intersection point close to the location.

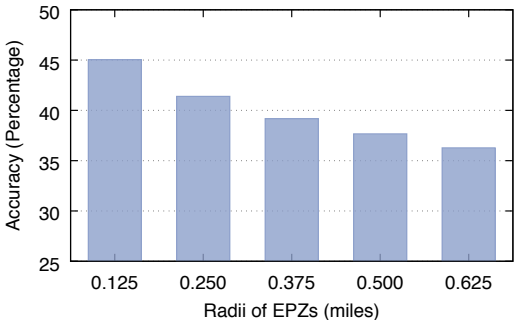


Figure 12: Efficacy of *Spatial Cloaking* defense (using different EPZ radii) against linear interpolation attacks.

the fuzzing value F by some constant factor:

$$\tau'_d = \tau_d + cF \tag{6}$$

We parameterized c by selecting a random subset of 1,000 athletes and running our algorithm using different c values but with a fixed F of 80 meters. As shown in Figure 10a, the optimal value of c turned out to be 1.

Having parameterized the attack, we next set out to tune our fuzzing parameter in order to identify an acceptable tradeoff between privacy and usability of the fitness tracking service. Selecting a different random subset of 1000 users, we applied the enhanced EPZ mechanism. For each of the 5 permissible Strava radii r, we applied different values of F ranging from 40 to r, with a ceiling of 500 meters. Several interesting findings emerge from our results, shown in Figure 10b. The first is that, while a protected location can be predicted with 96% accuracy when $r = 0.250$ miles, that accuracy drops to 32% with $r = 0.250$ miles and $F = 40$ meters. This is significant because a much larger section of the route is visible in the latter case in spite of the dramatically improved privacy level. It is also visible that higher F values quickly offer diminishing returns on privacy. At $F = 200$ meters (0.124 miles), accuracy is less than or equal to 15% against all radii. This validates our theory that injecting a small amount of noise into EPZ intersection points may lead to dramatic increases in privacy level. However, we note that there are likely more expressive models for the attacker to overcome fuzzing noise, which we leave for future work.

Spatial Cloaking Against this obfuscation, it no longer makes sense for an attacker to predict the center of the enhanced EPZ, as the protected location is equally likely to fall anywhere within the circle. However, we predict that the direction of an activity route as it enters the EPZ still leaks significant information about the user's protected location. To demonstrate this, we propose a new attack that interpolates the direction of routes as they enter the EPZ. Figure 11 demonstrates the intuition of this approach. For each user activity, we inspect the last 2 GPS points at the end of the route, then extend the route through the EPZ with simple linear interpolation. After doing this for every activity, we tabulate all of the points in the EPZ at which these lines intersect. We then group these intersections together to find the maximum number of intersection points that fall within t_d of one another. If multiple intersection points were found that fell within t_d of each other, we calculated the centroid of these points and issued a prediction. We considered our prediction successful if the highest confidence centroid fell within 50 meters of the actual protected location.

Radii	Random Guess	Prediction	Improvement
0.125	6.178%	45.0 %	7x
0.250	1.544%	41.3 %	27x
0.375	0.686%	39.1 %	57x
0.500	0.386%	37.6 %	98x
0.625	0.247%	36.2 %	147x

Table 2: Success rate of our attack on spatial cloaking compared to randomly guessing. Although the obfuscation reduces our identification rate, our attack significantly outperforms chance levels.

Our results can be found in Table 2. *Unsettlingly, this simple interpolation attack is 36.2 % - 45.0 % accurate against geo-indistinguishability techniques.* To demonstrate the significance of this result, consider the likelihood of predicting the protected location by issuing a random guess that falls within the EPZ, as shown in Table 2. For small privacy zones, our approach offers a 7x improvement over random guess; against large privacy zones, our approach offers a 147x improvement over random guessing. We also obtained similar results when running our fuzzing obfuscation against the interpolation attack. While the identification rate here is still low, it is not difficult to imagine that a more sophisticated version of this attack that leverages more expressive interpolation techniques and incorporates map information to reduce the search space. These results point to a natural tension between the desire to publish route information while concealing sensitive endpoints; significant amounts of private information is leaked through inspecting the trajectory of the route. At the same time, this countermeasure significantly increases the complexity of breaking an EPZ, which may prove sufficient to dissuade attackers in practice.

7 Discussion & Mitigation

7.1 Strava's Global Heat Map Incident.

The release of Strava's Global Heatmap published aggregated public usage data for 27 million users [14]. The motivation for publishing the heatmap was to help provide a resource for athletes to explore and discover new places to exercise; in addition, a related *Strava Metro* project leveraged this heatmap data to assist departments of transportation and city planning groups in improving infrastructure for bicyclists and pedestrians [19]. However, as a result of the sparsity of background noise in some regions, the heatmap was observed to leak sensitive and classified information regarding the locations of military bases, covert black sites and patrol routes, to name a few [24]. This information which could be turned into actionable intelligence, leading to potentially life-threatening situations [46].

Following the news coverage of privacy leakage in the global heatmap, we became curious about the privacy habits of the Strava users that exercised at these facilities. We searched our dataset for activities from three of the locations identified in popular media: the United Kingdom's Government Communications Headquarters (GCHQ), Australia's Pine Gap military facility, and Kandahar Airforce Base in Afghanistan. We found that 1 of 7 athletes in our dataset were using EPZs at GCHQ, 1 of 8 athletes used EPZs at Pine Gap, and 1 of 13 athletes used EPZs at Kandahar, suggesting that a non-negligible minority of athletes at these sites were aware of the privacy risks and were attempting to safeguard their usage.

The findings presented in this study potentially exacerbate the safety risks posed by the global heatmap revelations. Because many of the discovered facilities are highly secure, their identification in the heatmap may not pose an immediate threat to the safety of personnel. However, while the identities of specific athletes were not directly leaked in the heatmap, a related vulnerability allows an attacker to upload spoofed GPS data in order to discover the IDs of Athletes in a given area [25]. They can then search Strava for off-site areas that the targeted athlete frequents, making EPZs the last line of defense for protecting the target's home. Unfortunately, we have demonstrated that EPZs (as originally implemented) are inadequate, meaning that, conceivably, an attacker could have used our technique to identify an insecure location associated with military or intelligence personnel. We note again that such an attack is presently much more difficult on Strava following updates to their EPZ mechanism, which we describe in Section 9.

7.2 Attack Replication.[9]

The implications of our EPZ Identification Attack extend beyond one single fitness tracking app. To demonstrate, we replicated our attack on Map My Tracks [18] and Garmin Connect [12].

Map My Tracks. Users can set EPZs of radii 500, 1000, or 1500 meters. Map My Tracks also permits users to export GPS coordinates of the activities of any user in a CSV format. Like Strava, it is possible to detect the presence of an EPZ by inspecting the "distance from start" value of the GPS coordinates, which does not start from 0 if a route began within an EPZ. We created an account on Map My Tracks and uploaded 4 activities starting from the same "sensitive" location. Regardless of the EPZ size used, we successfully identified the sensitive location by running our attack, We did not need to reparameterize our algorithm (i.e., τ_d, τ_i), indicating that our values are robust across multiple services.

[9] Here, we describe an attack replication on companies' prior EPZ mechanisms, which were modified following vulnerability disclosure.

Garmin Connect. Garmin Connect is fitness tracking services that allow users to share activities tracked with compatible Garmin devices. Garmin Connect provides EPZs with radii ranging from 100 to 1000 meters in 100 meter increments. Like Map My Tracks, Garmin Connect allows users to export GPS coordinates of activities of other users in GPX format (a light-weight XML data format). Here, discrepancies between the route information and advertised distance once again makes it possible to infer when an EPZ is enabled on an activity. Creating an account on Garmin Connect, we uploaded 3 activities starting from a "sensitive" location. When launching our attack against 100, 500, and 1000 meter EPZs, we reliably recovered the protected location.

7.3 Additional Mitigations

In addition to the specific privacy enhancements presented above, we also advise fitness tracking services to adopt the following general countermeasures to order to increase the difficulty of abusing their services:

Randomize Resource IDs. Strava and Map My Tracks use sequential resource identifiers; data resources identifiers should be randomly assigned from a large space of possible identifiers (e.g., 2^{64}), as already done by Garmin Connect, to prevent the bulk enumeration of resources.

Authenticate All Resource Requests. Strava facilitates surveillance at scale because it does not require authentication in order to access resources. To address this concern, we recommend placing fine-grained resources behind an authentication wall so that Strava can monitor or suspend accounts that issue a high volume of requests.

Server-Side Rendering of Map Resources. We do not believe that it is necessary to expose raw GPS coordinates to the client in order to provide an enriched user experience. Instead, activity maps could be rendered at the server, or at least filtered and fuzzed to frustrate EPZ location attempts.

Conceal Existence of EPZ. Route information exposed to clients should be consistent in the claims they make about the length of routes. The advertised distance of an activity should be modified to reflect the portion of the route that is hidden by the EPZ. Had there been consistency of distance claims in our study, we would have been unable to obtain a ground truth as to whether or not an EPZ was enabled on the activity. While our methodology could still be used to detect likely EPZs in the absence of ground truth, there would also be a large number of false positives resulting from attempting to look for EPZs where they did not exist.

8 Related Work

Prior to this study, the privacy considerations of fitness apps has received little consideration in the literature. Williams [11] conducted a detailed study of Strava users and their behavior towards Strava application. He concluded that the majority of participants had considered privacy issues when using the application and had taken some measures to protect themselves, such as setting up privacy zones or not listing their equipment. However, in this work we show that only 9% of all the activities we studied were using privacy zones, calling this result into question. Further, we demonstrated that the privacy measures provided by Strava are insufficient to protect user privacy. The demographics of Strava users [4] indicate that an attacker would have an ample supply of potential targets to choose from; as seen in [6, 17], property theft against Strava users has already been reported in the media. Our findings provide a viable explanation for how these attacks could occur.

8.1 Location Privacy

Geo-indistinguishability has been used previously [30, 55] to provide static location privacy by perturbing the real location with fake location. Geo-indistinguishability is derived from differential privacy [35] and ensures that for any two location that are geographically close it will produce a pseduo-location with similar probabilities. Andrés *et al*. [26] used Planar Laplace mechanism to achieve ε geo-indistinguishability by using noise drawn from a polar Laplacian distribution and added to real locations. However, these techniques are not directly applicable to mobility data such as athletes routes that we consider in this paper. Existing work on mobility-aware location obfuscation technique [29] replaces real location traces with plausible fake location traces using human mobility model. However, this technique cannot be used directly in the context of fitness tracking apps as users still want to share a major portion of a route while preserving a certain portion of route (e.g. home).

In some instances, prior work has demonstrated applicable techniques for Preserving endpoint privacy while sharing route data. Duckham and Kulik [34] present location obfuscation techniques for protecting user privacy by adding dummy points in measurements with the same probability as the real user position. Ardagna et al. [27] demonstrate how an EPZ can be used to obfuscate users locations in order to preserve privacy, although possible weaknesses in this method are raised in [52]. In this work, we have demonstrated proof-of-concept attacks that can violate user privacy even in the presence of these obfuscations.

8.2 Social Network Privacy

The social network aspect of fitness tracking services allows users to "follow" each other, giving them access to additional data about each other. This can lead to social engineering [39, 5] and even automated social botnet attacks as in [28, 31], where user information such as location is automatically extracted. Strava provides a privacy option to require user approval for new followers, we show that when this option is not enabled such attacks are also possible on Strava and other fitness apps. A variety of privacy vulnerabilities have been identified on other social network platforms, ranging from server-side surveillance [33], third party application spying [54], and profiling of personality types [51]. This study confirms that a number of these concerns are also present in fitness tracking social networks.

8.3 Mobile Privacy

The functionality of fitness tracking social networks is predicated on the ubiquity of modern smart phones equipped with GPS and other private information (e.g., sensor readings). Lessons learned in the security literature regarding mobile application permissions could also be applied in the fitness space to improve user privacy. Enck *et al*. demonstrate a method of detecting application leakage of sensor information on the Android platform through taint analysis [36], and subsequently conducted a semi-automated analysis of a corpus of 1,100 applications in search of security and privacy concerns [37]. Felt *et al*. conduct a survey of application privileges and discovered that one-third of Android apps requested privileges that they did not need [38]. Our work suggests that overprivilege may also be a concern for third party applications that interoperate with fitness apps.

9 Ethics and Disclosure

Given the potential real-world privacy implications of this study, we have taken a variety of steps to ensure our research was conducted responsibly. We have consulted our Institutional Review Board (IRB) to confirm that our analysis of social media posts does not meet the definition of human subjects research (as defined in 45CFR46(d)(f) or at 21CFR56.102(c)(e)) and thus does not require IRB approval. The rationale provided was that analysis of public datasets such as social media posts does not constitute human subjects research. We note that our use of social media posts is consistent with prior research on user privacy [42, 56, 45, 53, 48], particularly studies that have evaluated location privacy and user discovery [47, 43, 49].

We have disclosed our findings to Strava, Garmin Connect, and Map My Tracks. As of the date of publication, all three companies have acknowledged the vulnerability and have incorporated one or more of our recommended countermeasures into their production systems. Strava has adopted a spatial cloaking function that is invoked upon the creation of every new user-specified EPZ, and provides the user with an option of re-randomizing the EPZ if they do not like its placement. Additionally, Strava has taken steps to prevent the bulk collection of their public user activities, including aggressive rate limiting of the `strava.com/stream/` API, least privilege restrictions on returned API fields based on the client's authorization state, and IP whitelisting of interoperable social network's servers to prevent unauthorized use of other APIs. Garmin Connect has introduced a randomization step similar to our EPZ intersection fuzzing technique – each time a new activity crosses an EPZ, the point at which the route is truncated is perturbed according to a random distribution. Additionally, Garmin Connect has added an optional user-driven obfuscation when a user attempts to create an EPZ, they may now drag the EPZ center away from their house, and moreover a message has been added to encourage users to set up multiple overlapping privacy zones. Map My Tracks also reported that they incorporated spatial cloaking into their new EPZ feature, but declined to discuss the details of their solution.

10 Conclusion

As fitness tracking services have grown in popularity, the online sharing of fitness data has created concerns for personal privacy and even national security. Understanding the effectiveness of privacy protections in such a system is paramount. In this paper, we have conducted a deep analysis of the privacy properties of Strava, an exemplar fitness tracking app. While we identified significant demand for privacy protections by users of these services, we have also demonstrated current mechanisms are inadequate – we found that the homes privacy-conscious athletes are consistently identifiable by attackers, and in fact that the only truly safe athletes are those that use the service infrequently. Through the insights gained in this study, we were able to develop and empirically demonstrate the efficacy of several novel privacy mechanisms that have been put into practice by major fitness tracking services. It is our hope that this work spurs greater interest in the efficacy and usability of privacy features in fitness tracking apps.

Acknowledgments

We would like to thank Adam Aviv for his valuable comments on an early draft of this paper. We also thank the anonymous reviewers for their helpful feedback. This work was supported in part by NSF CNS grants 16-57534 and 17-50024. The views expressed are those of the authors only.

References

[1] mapmyride. http://www.mapmyride.com/.

[2] Data Driven: Strava Users By The Numbers. http://www.triathlete.com/2016/04/features/data-driven-strav_130658.

[3] Fitbit. https://www.fitbit.com/.

[4] How Strava Is Changing the Way We Ride. https://www.outsideonline.com/1912501/how-strava-changing-way-we-ride.

[5] Strava, popular with cyclists and runners, wants to sell its data to urban planners. http://blogs.wsj.com/digits/2014/05/07/strava-popular-with-cyclists-and-runners-wants-to-sell-its-data-to-urban-planners/.

[6] Ride mapping sites: The bike thief's new best friend? http://www.cyclingweekly.co.uk/news/comment/ride-mapping-sites-the-bike-thiefs-new-best-friend-44149.

[7] Nike+. http://www.nike.com/us/en_us/c/nike-plus.

[8] Mining the Strava data. http://olivernash.org/2014/05/25/mining-the-strava-data/.

[9] Data Mining Strava. http://webmining.olariu.org/data-mining-strava/.

[10] strava-data-mining. https://github.com/wmycroft/strava-data-mining.

[11] King of the Mountain: A Rapid Ethnography of Strava Cycling. https://uclic.ucl.ac.uk/content/2-study/4-current-taught-course/1-distinction-projects/4-2013/williams-2012.pdf.

[12] Garmin Connect. https://connect.garmin.com/.

[13] Garmin Adds Privacy Zones for Public Activities. http://myitforum.com/myitforumwp/2017/04/12/garmin-adds-privacy-zones-for-public-activities/.

[14] Strava Global Heatmap - Strava Labs. http://labs.strava.com/heatmap/.

[15] Privacy Zones. https://support.strava.com/hc/en-us/articles/115000173384.

[16] Hide sensitive locations with privacy zones. http://www.mapmytracks.com/blog/entry/hide-sensitive-locations-with-privacy-zones.

[17] Strava and stolen bikes. https://www.bikehub.co.za/forum/topic/166972-strava-and-stolen-bikes/.

[18] Map My Tracks. http://www.mapmytracks.com/.

[19] What is Strava Metro? https://support.strava.com/hc/en-us/articles/216918877-What-is-Strava-Metro-?

[20] endomondo. https://www.endomondo.com/.

[21] RunKeeper. https://runkeeper.com/.

[22] Runtastic: Running, Cycling and Fitness GPS Tracker. https://www.runtastic.com/.

[23] Strava — Run and Cycling Tracking on the Social Network for Athletes. https://www.strava.com/.

[24] U.S. soldiers are revealing sensitive and dangerous information by jogging. http://wapo.st/2BDFrA4.

[25] Advanced denanonymization through strava. http://steveloughran.blogspot.co.uk/2018/01/advanced-denanonymization-through-strava.html.

[26] ANDRÉS, M. E., BORDENABE, N. E., CHATZIKOKOLAKIS, K., AND PALAMIDESSI, C. Geo-indistinguishability: Differential privacy for location-based systems. In *CCS* (2013), ACM.

[27] ARDAGNA, C. A., CREMONINI, M., DAMIANI, E., DI VIMERCATI, S. D. C., AND SAMARATI, P. Location privacy protection through obfuscation-based techniques. In *IFIP Annual Conference on Data and Applications Security and Privacy* (2007), Springer.

[28] BILGE, L., STRUFE, T., BALZAROTTI, D., AND KIRDA, E. All your contacts are belong to us: automated identity theft attacks on social networks. In *WWW* (2009), ACM.

[29] BINDSCHAEDLER, V., AND SHOKRI, R. Synthesizing plausible privacy-preserving location traces. In *IEEE Symposium on Security and Privacy* (2016), IEEE.

[30] BORDENABE, N. E., CHATZIKOKOLAKIS, K., AND PALAMIDESSI, C. Optimal geo-indistinguishable mechanisms for location privacy. In *CCS* (2014), ACM.

[31] BOSHMAF, Y., MUSLUKHOV, I., BEZNOSOV, K., AND RIPEANU, M. The socialbot network: when bots socialize for fame and money. In *Proceedings of the 27th annual computer security applications conference* (2011), ACM.

[32] CHERNOV, N., AND LESORT, C. Least squares fitting of circles. *Journal of Mathematical Imaging and Vision 23*, 3 (Nov 2005), 239–252.

[33] CRISTOFARO, E. D., SORIENTE, C., TSUDIK, G., AND WILLIAMS, A. Hummingbird: Privacy at the time of twitter. In *IEEE Symposium on Security and Privacy* (2012).

[34] DUCKHAM, M., AND KULIK, L. A formal model of obfuscation and negotiation for location privacy. In *International Conference on Pervasive Computing* (2005), Springer, pp. 152–170.

[35] DWORK, C. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation* (2008), Springer, pp. 1–19.

[36] ENCK, W., GILBERT, P., CHUN, B.-G., COX, L. P., JUNG, J., MCDANIEL, P., AND SHETH, A. N. TaintDroid: An Information-flow Tracking System for Realtime Privacy Monitoring on Smartphones. In *OSDI* (Oct. 2010).

[37] ENCK, W., OCTEAU, D., MCDANIEL, P., AND CHAUDHURI, S. A Study of Android Application Security. In *Proceedings of the 20th USENIX Security Symposium* (2011).

[38] FELT, A. P., CHIN, E., HANNA, S., SONG, D., AND WAGNER, D. Android Permissions Demystified. In *CCS* (2011), ACM.

[39] FRÖHLICH, S., SPRINGER, T., DINTER, S., PAPE, S., SCHILL, A., AND KRIMMLING, J. Bikenow: a pervasive application for crowdsourcing bicycle traffic data. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct* (2016), ACM, pp. 1408–1417.

[40] GANDER, W., GOLUB, G. H., AND STREBEL, R. Least-squares fitting of circles and ellipses. *BIT Numerical Mathematics 34*, 4 (1994), 558–578.

[41] GRUTESER, M., AND GRUNWALD, D. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of the 1st international conference on Mobile systems, applications and services* (2003), ACM, pp. 31–42.

[42] HU, H., AHN, G.-J., AND JORGENSEN, J. Detecting and resolving privacy conflicts for collaborative data sharing in online social networks. In *ACSAC* (2011), ACM.

[43] LI, M., ZHU, H., GAO, Z., CHEN, S., YU, L., HU, S., AND REN, K. All your location are belong to us: Breaking mobile social networks for automated user location tracking. In *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing* (2014), ACM, pp. 43–52.

[44] LUPTON, D. *You are Your Data: Self-Tracking Practices and Concepts of Data*. Springer Fachmedien Wiesbaden, Wiesbaden, 2016, pp. 61–79.

[45] MAO, H., SHUAI, X., AND KAPADIA, A. Loose tweets: An analysis of privacy leaks on twitter. In *Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society* (2011), WPES '11, ACM.

[46] MCDONOUGH, J. Strava has Data that Most Intelligence Entities Would Literally Kill to Acquire. http://news.theceomagazine.com/technology/strava-data-intelligence-entities-literally-kill-acquire/.

[47] POLAKIS, I., ARGYROS, G., PETSIOS, T., SIVAKORN, S., AND KEROMYTIS, A. D. Where's wally?: Precise user discovery attacks in location proximity services. In *CCS* (2015), ACM.

[48] PUTTASWAMY, K. P., AND ZHAO, B. Y. Preserving privacy in location-based mobile social applications. In *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications* (2010), ACM, pp. 1–6.

[49] QIN, G., PATSAKIS, C., AND BOUROCHE, M. Playing hide and seek with mobile dating applications. In *IFIP International Information Security Conference* (2014), Springer, pp. 185–196.

[50] QUARLES, J. A Letter to the Strava Community. https://blog.strava.com/press/a-letter-to-the-strava-community/.

[51] QUERCIA, D., KOSINSKI, M., STILLWELL, D., AND CROWCROFT, J. Our twitter profiles, our selves: Predicting personality with twitter. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing* (Oct 2011), pp. 180–185.

[52] SRIVATSA, M., AND HICKS, M. Deanonymizing mobility traces: Using social network as a side-channel. In *CCS* (2012), ACM.

[53] VICENTE, C. R., FRENI, D., BETTINI, C., AND JENSEN, C. S. Location-related privacy in geo-social networks. *IEEE Internet Computing 15*, 3 (May 2011), 20–27.

[54] WANG, N., XU, H., AND GROSSKLAGS, J. Third-party apps on facebook: Privacy and the illusion of control. In *Proceedings of the 5th ACM Symposium on Computer Human Interaction for Management of Information Technology* (2011), CHIMIT '11, ACM.

[55] YU, L., LIU, L., AND PU, C. Dynamic differential location privacy with personalized error bounds. In *NDSS* (2017).

[56] ZHANG, C., SUN, J., ZHU, X., AND FANG, Y. Privacy and security for online social networks: challenges and opportunities. *IEEE Network 24*, 4 (July 2010), 13–18.

[57] ZHU, J. Conversion of earth-centered earth-fixed coordinates to geodetic coordinates. *IEEE Transactions on Aerospace and Electronic Systems 30*, 3 (1994), 957–961.