A Run a Day Won't Keep the Hacker Away: Inference Attacks on Endpoint Privacy Zones in Fitness Tracking Social Networks

Karel Dhondt
imec-DistriNet, KU Leuven
Ghent, Belgium
karel.dhondt@kuleuven.be

Victor Le Pochat
imec-DistriNet, KU Leuven
Leuven, Belgium
victor.lepochat@kuleuven.be

Alexios Voulimeneas
imec-DistriNet, KU Leuven
Ghent, Belgium
alex.voulimeneas@kuleuven.be

Wouter Joosen
imec-DistriNet, KU Leuven
Leuven, Belgium
wouter.joosen@kuleuven.be

Stijn Volckaert
imec-DistriNet, KU Leuven
Ghent, Belgium
stijn.volckaert@kuleuven.be

ABSTRACT

Fitness tracking social networks such as Strava allow users to record sports activities and share them publicly. Sharing encourages peer interaction but also constitutes a risk, because an activity's start or finish may inadvertently reveal privacy-sensitive locations such as a home or workplace. To mitigate this risk, networks introduced *endpoint privacy zones* (EPZs), which hide track portions around protected locations. In this paper, we show that EPZ implementations of major services remain vulnerable to inference attacks that significantly reduce the effective anonymity provided by the EPZ, and even reveal the protected location. Our attack leverages distance information leaked in activity metadata, street grid data, and the locations of the entry points into the EPZ. This yields a constrained search space where we use regression analysis to predict protected locations. Our evaluation on 1.4 million Strava activities shows that our attack discovers the protected location for up to 85% of EPZs. Larger EPZs reduce the performance of our attack, while geographically dispersed activities in sparser street grids yield better performance. We propose six countermeasures, that, however, come with a usability trade-off, and responsibly disclosed our findings and countermeasures to the major networks.

CCS CONCEPTS

• **Security and privacy** → **Privacy protections**; **Usability in security and privacy**; **Social aspects of security and privacy**; **Social network security and privacy**.

KEYWORDS

privacy zones; fitness tracking social networks; location privacy; location-based services; privacy

ACM Reference Format:
Karel Dhondt, Victor Le Pochat, Alexios Voulimeneas, Wouter Joosen, and Stijn Volckaert. 2022. A Run a Day Won't Keep the Hacker Away: Inference Attacks on Endpoint Privacy Zones in Fitness Tracking Social Networks. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22), November 7–11, 2022, Los Angeles, CA, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3548606.3560616

This is an extended version that includes the appendices.

CCS '22, November 7–11, 2022, Los Angeles, CA, USA

ACM ISBN 978-1-4503-9450-5/22/11.
https://doi.org/10.1145/3548606.3560616

1 INTRODUCTION

Fitness tracking social networks (FTSNs) consistently rank among the most popular mobile apps and saw an additional surge in popularity during the COVID-19 pandemic [4, 49]. For example, one of the largest networks, Strava, has over 100 million registered users [50]. These fitness tracking social networks allow users to record their sports activities, and share their tracks and achievements with friends and other users of the platform, promoting enjoyment and motivation [7]. The tracks represent the routes that the user followed during the activity. While sharing tracks forms part of the attraction of these networks, this, however, comes with privacy and security risks, as they might reveal sensitive information, such as the user's regular routes or visited locations, to people with ill intentions. Several past incidents drew attention to the dangers of sharing this data with the public, from revealing secret military locations [25], enabling theft of exercise equipment [10], revealing the identity of nearby athletes [45], to doxing users [39].

To limit the potential risks of sharing information, all major networks offer privacy controls that limit the amount of shared information, as well as control whom information is shared with. One notable privacy control is the *endpoint privacy zone* (EPZ). An EPZ allows users to hide track portions near protected (sensitive) locations, such as their home or work address, from all activity tracks shown to other users. This measure aims to prevent harassment and stalking at commonly visited locations [54], or criminal activity such as theft at the protected locations [10]. Most commonly, EPZ implementations balance an increase in privacy with usability, notably in terms of tracking fitness achievements, as features such as leaderboards may only be available for publicly viewable data [14]. Moreover, activities usually still contain the full traveled distance, even for the portions hidden by the EPZ. The user who created the activity can also still access the full activity, as shown in Figure 1.

One implementation of EPZs hides track portions inside a circle with the sensitive location as its center and a user-configurable radius. However, this implementation has fallen out of favor, as given one or more tracks for one user, it is possible to reconstruct the privacy zone (i.e., the radius) and find the sensitive location

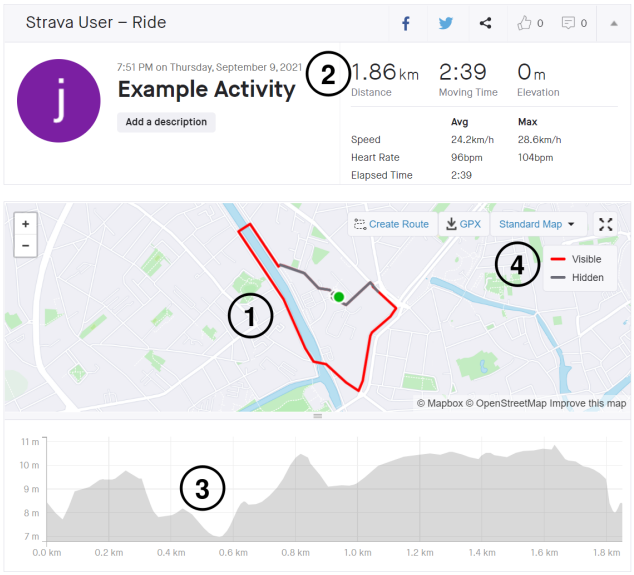


Figure 1: Example of an activity in Strava. The interface displays the track ①, the total traveled distance ② and accumulated distance at each point (through the elevation profile ③). Part of the track is marked as 'hidden' ④, as it is cloaked by an EPZ. Only the owner of the activity sees that part of the activity is hidden; other users only see the visible part without any indication that another part is hidden.

(i.e., the center) through basic geometric inference. In 2018, Hassan et al. showed that this EPZ implementation is vulnerable to automated inference attacks [24]. In 2022, Mink et al. showed that human users can visually infer privacy zones that use this implementation [35].

We confirm through a systematic analysis that major fitness tracking social networks implemented some countermeasures in an attempt to better protect sensitive locations from these basic inference attacks [24, 35]. These countermeasures include those proposed by Hassan et al. [24]: larger radii, noise addition at the EPZ boundary, and spatial cloaking where the EPZ center is randomly shifted. However, some usability trade-offs are made, such as allowing small or only fixed radii, or not requiring spatial cloaking. Moreover, some fitness tracking social networks still do not implement EPZs at all.

In this paper, we are the first to show that these newer implementations of EPZs remain highly vulnerable to the discovery of the purportedly protected location, even when countermeasures such as spatial cloaking are applied. Our novel inference attack leverages two inputs that enable a regression analysis resulting in the protected location for an EPZ. First, the road network restricts the possible paths that a user could have taken inside an EPZ. Second, the activity metadata leaks the exact distances of the paths that were traveled inside the EPZ. We then predict the protected location as the point where the distances of the possible and actual paths match best across multiple activities.

We find through an evaluation on 1.4 million real-world Strava activities that our attack can deanonymize protected locations for up to 85% of EPZs. Larger EPZ radii are more effective at preventing location inference and preserving user privacy, but even for very large radii (1 km), deanonymization remains possible for 55% of EPZs. Through a detailed analysis of the sensitivity of our attack, we find that higher geographic activity diversity and lower street density benefit its performance.

We propose and evaluate six countermeasures that can restore the anonymity of locations protected by EPZs to a varying degree. We find that generalization (rounding) of reported activity distances would be the most effective countermeasure, although it comes with a significant negative usability impact, as fitness tracking social networks are attractive precisely because they allow tracking small achievement differences accurately. Other countermeasures may be less invasive, but are then also less effective at improving privacy. Interestingly, certain interventions such as regenerating EPZs may actually improve our attack's efficacy, as they provide more data from which the protected location can be more reliably inferred. Countermeasures must therefore be carefully evaluated in order to minimize their negative impact on usability and privacy. We disclosed our findings and proposed countermeasures to the major vulnerable fitness tracking social networks.

In summary, we make the following contributions:

- We conduct a systematic analysis on current EPZ implementations for the most popular fitness tracking social networks.
- We develop a proof-of-concept attack that infers protected locations inside EPZs through regression analysis on the road network and leaked covered distances (Section 4).
- We evaluate our attack on 1.4 million real-world Strava activities (Section 5). We discover sensitive locations for up to 85% of EPZs, and find that our attack performs better with geographically dispersed activities on sparse street grids (Section 6).
- We propose six countermeasures to improve the anonymity provided by EPZs and discuss their trade-offs between privacy and usability (Section 7).

2 FITNESS TRACKING SOCIAL NETWORKS

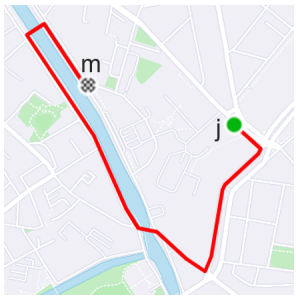
Our work concerns *fitness tracking social networks*, where users can record their workouts and share them with others. Users typically record workouts using a GPS-enabled smartphone or wearable device and upload them in the form of *activities*. An activity contains a *track*, i.e., the route that the user took, represented by a series of *points* (coordinates). The activity also reports the total distance traveled, as well as the accumulated distance at each point of the track, along with other metadata such as the duration, pace, elevation profile, heart rate, etc. Finally, many FTSNs define specific stretches of road (e.g., 'segments' on Strava) where they maintain a leaderboard of the fastest athletes across these stretches. Figure 1 shows what a fully developed activity may look like.

Most networks offer privacy controls that let users hide all activities from other users (i.e., private profile) or mark specific activities as hidden. Using these controls can severely limit the functionality of social features for the affected activities. For example, activities that are marked as private by their owners cannot be viewed by other users and may be ineligible for inclusion on segment leaderboards and in challenges [14, 7]. However, networks such as Strava

Table 1: Number of downloads from the Google Play Store and EPZ features of popular fitness tracking social networks.

Application	Downloads	EPZ	EPZ Radii (meter)
Adidas Runtastic [1]	50M+	✗	
Strava [48]	50M+	circular	[200, …, 1600] incr 200
Garmin Connect [21]	10M+	circular	[100,…, 1000] incr 100
Komoot [27]	10M+	polygon	
Map My Run [53]	10M+	✗	
Nike Run Club [38]	10M+	✗	
Relive [42]	10M+	circular	[200,…, 1000] incr 200
Ride With GPS [43]	1M+	circular	[150, 300, 600, 1200]
Map My Tracks [31]	100k+	circular	[500, 1000, 1500]

(a) Activity a ($\{p_1, \ldots, p_n\}$), as visible to its owner.



(b) Activity a' ($\{p_j, \ldots, p_m\}, 1 \leq j \leq m \leq n$), as visible to other users.

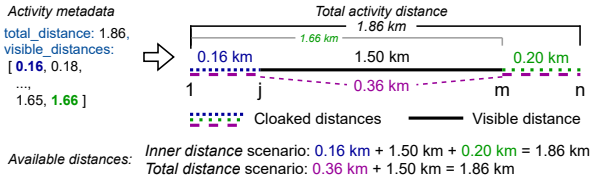


Figure 2: Visibility of an activity to which an EPZ has been applied. The EPZ is shown as a red circle.

mark activities as public by default [33] and the majority of users keep this setting [22].

(Endpoint) Privacy Zones (EPZs) are a more functionality-friendly privacy control and are supported by most networks. EPZs allow users to hide the most sensitive parts of the tracks in their activities, as they could otherwise reveal frequently visited locations such as the user's home or workplace. Activities with EPZs applied to them can still be shared, but the hidden parts of the tracks may still be ineligible for performance comparisons, so users may be inclined to disable or at least severely limit them. We discuss these zones in more detail in Section 3.

3 ENDPOINT PRIVACY ZONES

Most FTSNs allow their users to hide a privacy-sensitive location by letting them define an *Endpoint Privacy Zone* (EPZ) around that location. One location corresponds to exactly one EPZ, but a user can configure multiple locations that each have their own EPZ. EPZs can be regenerated at any time and are applied retroactively to all the activities of the owner. The goal of the EPZ is then to hide those parts of the track that might reveal the sensitive location, i.e., are near this location. Only the owner of the activity a can view the full track, including its *actual* start and finish points (together the 'endpoints'), as well as the layout of the EPZ itself (shown in Figure 2a). Other users only see a cloaked activity a' as defined in Definition 1 and depicted in Figure 2b. Concretely, in such a cloaked activity all points from the start of the track until the first time the owner leaves the EPZ are hidden, as are all points from the last time the owner enters the EPZ until the end of the track. The other users therefore observe *cloaked* start and finish points. Note that if the owner passes through the EPZ but does not start or end there, that track portion through the EPZ is not hidden. Moreover, even though the points inside the EPZ are hidden, the accumulated distance for points outside the EPZ as well as the total distance are not changed. This forms the basis for our attack described in Section 4.

Definition 1 (Endpoint Privacy Zone). Let protected location $p_s = (x_s, y_s)$ be a point in the Cartesian plane C, and a be an activity route of n points $\{p_1, \ldots, p_n\}$. We denote p_1 as the actual start point and p_n as the actual finish point.

Let EPZ be a subplane of C. Enforcing EPZ on activity a results in a cloaked activity $a' = \{p_j, \ldots, p_m\}$ with $1 \leq j \leq m \leq n$, where p_j is the first point and p_m the last point of the activity route that does not lie in EPZ. We denote p_j as the cloaked start point and p_m as the cloaked finish point.

In Table 1, we list the supported EPZ parameters of popular FTSNs. Most FTSNs use circular EPZs and let the user select the radius of the circle from a set of fixed radii. Hassan et al. inferred protected locations using publicly available information such as the advertised start and finish point of the user's activity [24]. The researchers demonstrated that, given multiple endpoints of protected activities and the circular layout of the EPZ, an adversary could reconstruct the EPZ and expose the protected location (i.e., the center point of the EPZ). To deal with the aforementioned attack, some FTSNs apply *spatial cloaking* by adding a random translation to the center of the EPZs [23], resulting in a cloaked circular EPZ as defined in Definition 2. An adversary could still determine the parameters of the cloaked EPZ, but cannot infer the protected location, since the protected location and the EPZ center do not match. Komoot uses a randomly shaped polygon around the protected location, rather than a circular EPZ. This makes it more difficult for an attacker to deduce the shape of the privacy zone [28]. Map My Tracks has an automatic privacy zone detection tool, lowering the bar for users to create a privacy zone [37]. This automatic tool scans all new activity endpoints in order to identify regular start and finish locations. If such locations are detected, a (circular) EPZ is enforced on all activities retroactively. This feature is enabled by default but can be disabled by the user at any time.

Definition 2 (Cloaked Circular Endpoint Privacy Zone). Let $cEPZ$ be a circle with a center $p_{s_{trans}}$ randomly translated from protected location p_s, and radius R. Enforcing $cEPZ$ on activity a results in an cloaked activity $a' = \{p_j, \ldots, p_m\}$ with $1 \leq j \leq m \leq n$, where p_j is the first point where $\text{dist}(p_j, p_{s_{trans}}) > R$, and p_m the last point where $\text{dist}(p_m, p_{s_{trans}}) > R$.

Figure 3: The intuition behind our attack: we search the protected location (shown as a black marker) as the point where the theoretical paths (based on reported distances, shown by dashed lines) starting from the EPZ entry gates (in different directions, shown by different colors) intersect.

4 BREAKING ENDPOINT PRIVACY ZONES

In this section, we present an attack against the current state-of-the-art EPZs under a predefined threat model. Crucially, this attack is enabled by the availability of exact accumulated and total distances for cloaked activities, revealing the distance traveled inside the EPZ. The intuition behind our attack is that if a victim has multiple activities in one or more cardinal directions (which we denote as 'entry gates'), we can use these distances to limit the locations where the victim could have started or ended their activities. For example, consider the three activities in Figure 3 indicated in blue, orange and green. While only the owner of the activities can see the dashed portions of the activities inside the EPZ, the attacker still knows their lengths. With these lengths and given the street grid, an adversary is able to construct all paths the victim could have covered inside the EPZ. Given these paths, the endpoint of the activities is at risk of being inferred using the intersection of multiple distinct paths inside the EPZ. We divide the breaking of privacy zones into two subproblems: first, identifying EPZs to reduce the search space, and second, finding the protected location inside those EPZs through regression analysis.

4.1 Threat Model

For this work, we consider an attacker who attempts to infer the protected locations of another user by only using the publicly available information that a regular user would be entitled to view, as displayed by the FTSN. The adversary has exactly the same rights as a regular user and cannot access any information by any other means, e.g., by infiltrating the FTSN's servers. The attacker can target either one specific user or the entire userbase of the network.

The attacker is reasonably technically sophisticated, capable of inspecting network traffic to retrieve the activity metadata (e.g., in browser developer tools), download map data and run the inference algorithm. This attacker model is similar to that of Hassan et al. [24]. Mink et al. [35] consider a less technically skilled attacker, who infers protected locations visually from the activity view on the network's website. In the context of intimate partner violence, Tseng et al. [52] found evidence of forum discussions where users collaborate on technically sophisticated attacks. Given possibly similar motivations to deanonymize protected locations (e.g., stalking), even less technically skilled attackers could receive support in successfully deploying the inference attack.

4.2 Identifying EPZs

As a preliminary step for our attack, we seek to identify the EPZs of a user. This step resembles the attack developed by Hassan et al. [24], where identifying the EPZ (by fitting a circle) reveals the protected location. However, this is no longer sufficient for our inference attack, as networks use EPZs with countermeasures against Hassan et al.'s attack. In our case, this step is also not strictly necessary for our attack to be effective, as we can search candidate locations throughout the entire street grid regardless of the EPZ layout. However, this step constrains the search space, which improves the attack's efficacy and computational performance by removing points that are ineligible as protected location. Moreover, if one search space contains multiple non-overlapping EPZs[1], our regression-based approach for location discovery (see Algorithm 2) may return a location outside these EPZs, i.e., in fact ineligible to be a protected location. Some restriction is therefore advantageous, although it does not need to be precise. Nevertheless, with knowledge of how EPZs are configured on a specific platform (see Section 3), the adversary can identify the distinct EPZs more precisely.

As an example of an EPZ identification algorithm, we present Algorithm 1 that discovers circular EPZs. Such EPZs are used by most services (Table 1). This algorithm identifies multiple circular EPZs, each with different protected locations and parameters, from the set A of all cloaked activities of one user. The input to our algorithm is the set P of all start points p_j for cloaked activities that started inside an EPZ and all finish points p_m for activities that ended in an EPZ, as they are shown to the adversary (a non-owner user)[2]. We then adapt the k-means clustering algorithm [30] to output the distinct circular EPZ layouts based on these start and finish points, iteratively increasing k until every EPZ is represented by exactly one cluster. For a given k, the algorithm initializes k clusters S_i $(i = 1, \ldots, k)$ with randomly selected points from P. In subsequent iterations, our algorithm assigns each point p to the cluster S_i with the lowest Euclidean distance to the least squared circle C_i fitted between the points of that cluster, and fitted such that none of the points lay inside C_i (a circular EPZ cloaks all points within the circle, cf. Definition 2). This cost function differs from regular k-means, where the distance from points to the mean of their cluster is minimized. The radius of the fitted circle should be bounded by the notion of the minimum and maximum radius of an

[1] In case the EPZs overlap, we consider the union as one EPZ with multiple protected locations. Bootstrapping the activities for the input to Algorithm 2 will then probabilistically return these two locations (see also Section 6.2).

[2] Note that we convert the geodetic coordinates (latitude and longitude) from the FTSN activity data, to plane coordinates (Universal Transverse Mercator) by projecting them onto a Cartesian plane, as established in Definition 1.

Algorithm 1 cEPZ identification algorithm

Input:
P ▷ Set of endpoints for cloaked activities
Output: $C_{1,\dots,k} = cEPZ_{1,\dots,k}$ ▷ k cEPZs

procedure cEPZ IDENTIFICATION(P)
  $k \leftarrow 1$
  do
    Initialize k clusters S_i[3] at random ($i = 1, \dots, k$)
    do
      ▷ Assignment step
      for point p in P **do**
        $i \leftarrow \text{argmin}_i \, \text{dist}(p, C_i)$
        $S_i = S_i \cup \{p\}$
      end for
      ▷ Update step
      for cluster S_i in S **do**
        $C_i \leftarrow$ LSQ fit circle through $\{p | p \in S_i\}$
      end for
    while $\exists C_i$: centroid change of $C_i > \tau_{converged}$
    $k \leftarrow k + 1$
  while $\exists p \in S_i : \text{dist}(p, C_i) > \tau_{disjoint}$
  return $C_{1,\dots,k}$
end procedure

EPZ implementation. This avoids too small clusters 'overfitted' on too few points such that multiple circles actually are part of one EPZ, and too large clusters due to outlier points.

Our algorithm repeats these assignments of points to clusters and updates to fitted circles until the circles' centroid changes between iterations are all lower than $\tau_{converged}$. This threshold should be (empirically) selected such that clusters are sufficiently stable while not indefinitely updating clusters. Moreover, k-means might return a local instead of global minimum because of the random assignment of points to clusters during the initialization phase. In practice, our algorithm will therefore repeat the initialization, assignment and update steps several times and use the clusters with the lowest distortion, i.e., the global squared sum of distances between points and the edge of their assigned fitted circle.

The algorithm finally checks whether all distinct EPZs have been identified, or whether multiple EPZs are still merged into one cluster. It therefore tests whether the maximum distance of every point p to its assigned fitted circle (EPZ) is lower than $\tau_{disjoint}$. This threshold could, for example, be the known maximum radius for the specific EPZ implementation. If this is not yet the case, we assume there is still a cluster containing multiple EPZs, and therefore increment k to add a cluster and restart at the random assignments of points to clusters. Once the condition on $\tau_{disjoint}$ is also met, the algorithm outputs the fitted circles $C_{1,\dots,k}$ through the points of each cluster S_i, which correspond to the k *cEPZ*s configured by the user.

4.3 Finding the Protected Location for an EPZ

The core of our attack consists of predicting protected locations for each individual EPZ previously identified by the adversary. We associate this EPZ with the subset of user activities A_{EPZ} that were cloaked using this EPZ (cf. Definition 1). We then retrieve the road network graph G_{EPZ} inside the EPZ, defined as a set of edges E through nodes V that represent all the possible protected locations. This effectively constrains our search space to a finite set of locations, and therefore reduces the identification of the protected location inside an EPZ from a continuous to a discrete problem. The correctness of our solution is therefore also limited by the resolution of the graph G. This resolution can be improved using *chaining*, i.e., adding equidistant, intermediate nodes at a certain interval distance d_{chain}, e.g., selected to mimic GPS precision, on edges longer than d_{chain} as depicted in Figure 4b.

[3]with corresponding circles C_i

Algorithm 2 Protected location prediction algorithm

Input:
$G_{EPZ} = (V, E)$ ▷ Road graph inside EPZ
$A_{EPZ} = (P, O)$ ▷ Endpoints and distances inside EPZ
Output: $v_{protected}$ ▷ Predicted protected location

procedure PREDICT PROTECTED LOCATIONS(A, G)
  ▷ Calculate theoretical distances
  $A' \leftarrow \emptyset$
  for pair (p_l, o_l) in A **do**
    $v_{p_l} \leftarrow \text{argmin}_{v \in V} \, \text{dist}(p_l, v)$
    $d_{p_l} \leftarrow \text{dist}(p_l, v_{p_l})$
    if $d_{p_l} \leq \tau_{snap}$ **then**
      $A' \leftarrow A' \cup \{(v_{p_l}, o_l)\}$
    end if
  end for
  $T \leftarrow |A'| \times |V|$ matrix
  for pair (v_{p_l}, o_l) in A' **do**
    $T_{v_{p_l},*} \leftarrow \text{dijkstra\_single\_source\_lengths}(v_{p_l}, G)$
  end for
  ▷ Identify entry gates Y
  $Y \leftarrow DBSCAN(A', \epsilon, minPts)$
  ▷ Remove deviating activities
  $A'' \leftarrow \emptyset$
  for entry gate $Y_i = (P_i, O_i) \subseteq A'$ in Y **do**
    $Y_i' \leftarrow \{(v_{p_l}, o_l) | (v_{p_l}, o_l) \in Y_i, o_l \leq \max(T_{l,*})\}$
    $Y_i'' \leftarrow \{(v'_{p_l}, o'_l) | (v'_{p_l}, o'_l) \in Y_i', |o'_l - \overline{O'_i}| \leq 3\sigma_{O'_i}\}$
    $A'' \leftarrow A'' \cup Y_i''$
  end for
  ▷ Predict protected location
  $v_{protected} \leftarrow \text{argmin}_{v_l \in V} \sum_{(v''_{p_l}, o''_l) \in A''} \left| o''_l - T_{v''_{p_l}, v_l} \right|$
  return $v_{protected}$
end procedure

We propose Algorithm 2 that predicts the most probable candidate based on two inputs. The first input is the previously constructed road graph G with nodes V and edges E. The second input is the set of cloaked activities A_{EPZ}, as defined by a mapping from the union of their cloaked start (p_j) and finish (p_m) points P, to the reported distances O between the actual and cloaked start points ($\text{dist}(p_1, p_j)$) or the actual and cloaked finish points ($\text{dist}(p_m, p_n)$), respectively. These reported distances are available through the activity data (e.g., elevation profile). In our *inner distance* scenario, the distances for the start and finish point are available separately, as the accumulated distance from the beginning at the cloaked start

point and the remaining distance until the end from the cloaked finish point respectively. We generalize this to a *total distance* scenario in which only the combined cloaked distance is available, i.e., without distinct distances for the start and finish point. If only one of the points lies inside a privacy zone (as could be inferred using Algorithm 1), the total distance trivially reduces to the inner distance, as this inner distance at the single cloaked side is equal to the reported distance discrepancy. Our data characteristics in Section 5.2 show that 35.08% of activities only start or finish inside an EPZ, but not both. When both endpoints are cloaked, there is an unresolvable degree of freedom in the division of the distance discrepancy over the start and finish points. In this case, the activity is discarded, and the attack is run on the remaining activities. Our analysis in Section 6.3 shows that our attack still performs well in this total distance scenario, with only a minor performance reduction compared to the inner distance scenario.

In an ideal context, the endpoints for the reported distances of all activities would overlap at exactly one node of the road graph (i.e., the protected location) if the following idealistic assumptions were to hold:

A1. The cloaked start and finish points intersect with the edge of the EPZ.
A2. A protected location is always located along a path, i.e., on a node inside G.
A3. The victim starts and finishes their activity at the single protected location inside a distinct EPZ.
A4. Inside the EPZ, the victim uses the shortest path on the street grid from or to the protected location.

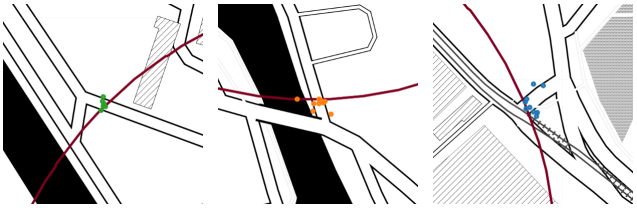
However, in practice these assumptions do not always hold. For example, a user might only start recording their activity 50 meters away from their protected location, and GPS tracking errors will cause a track to deviate from (the shortest path on) the street grid. Overall, for a 200m EPZ, 54% of all activities in our real-world data set (Section 5) violate at least one idealistic assumption. Our approach is explicitly designed to be robust against these violations, i.e., it works even in non-ideal settings. We develop a four-step algorithm that provides a sufficiently correct solution even though these assumptions from the ideal context do not hold, as we show in our real-world evaluation (Section 6).

4.3.1 Calculate Theoretical Distances. The algorithm starts by map matching ('snapping') all endpoints to the road graph. For each point $p_l \in P$, the algorithm identifies the node $v_{p_l} \in V$ with the lowest Euclidean distance d_{p_l} to p_l. If this distance exceeds a threshold τ_{snap}, the algorithm discards this point since it is uncertain if the road was used. τ_{snap} could be empirically selected based on the mean GPS sampling distance. P' represents the set of map matched points v_{p_l} that are retained. The algorithm then computes the shortest path lengths from v_{p_l} (recall: a point near the edge of the EPZ) to all other nodes V inside the EPZ using the Dijkstra single source multiple destination algorithm [16]. The lengths are collected into a distance matrix T of size $|P'| \times |V|$. These lengths represent the 'theoretical' distances from the cloaked map matched endpoints at the edge of the EPZ to nodes within the EPZ (i.e., possible protected locations), if the track were to exactly follow the road graph. However, GPS errors and the variation across the width of a road cause the actual user tracks to deviate from this road graph. The regression analysis in the last step of our algorithm addresses these deviations when predicting the protected location.

(a) Example road network graph G for one EPZ.

(b) Road network with node resolution increased through chaining.

Figure 4: The road network graph G constrains the search space for our location prediction algorithm. Each node $v \in V$, displayed in red, is a potential protected location.



(a) (b) (c)

Figure 5: Entry gates are identified as clusters of close endpoints. Endpoints will not exactly intersect the circular EPZ, as the first (last) point outside the EPZ will be the first (last) visible point. No endpoints therefore lie inside the EPZ.

4.3.2 Identify Entry Gates. The exact intersection point of an activity and the EPZ is seldom recoverable, violating Assumption A1. Definition 1 states that the activity gets cut off at the first point outside the EPZ rather than at the exact intersection. In combination with low GPS sampling rates (to save battery) and GPS errors, this cloaked endpoint can be distant from the edge of the EPZ, particularly as the speed of an activity increases. Figure 5 shows an example of this scenario, where the endpoints of an activity do not exactly intersect with the circular EPZ.

We cluster the cloaked endpoints p_l to form dense regions. We refer to these regions as *entry gates*, and we assume that each entry gate Y_j consists of endpoints $P_{Y_j} \subseteq P'$ where the intersection point of the actual track with the edge of the EPZ is shared. In Figure 5, the points that form one entry gate are displayed in the same color. The most suitable algorithm for geospatial data is DBSCAN [19], a density-based clustering algorithm, since it does not require a priori knowledge of the number of clusters (i.e., entry gates). DBSCAN requires two parameters: ϵ, the maximum distance between two points of the cluster, and *minPts* the minimum number of points to form a dense region. For example, ϵ could be determined as the 95th

quantile of sample distances (i.e. distances between consecutive activity points). We run DBSCAN on the points P' to obtain all entry gates Y, each defined by a subset $P_{Y_j} \subseteq P'$ of the points corresponding to that entry gate, disjoint with all other subsets.

4.3.3 Remove Deviating Activities. Next, we discard outlier endpoints that could incorrectly skew the prediction of the protected location. Such outliers may stem from activities where the user started or finished far away from the protected location (violating Assumption A3), or did not follow the shortest path within the EPZ (violating Assumption A4).

The algorithm starts by discarding points v_{p_l} with a reported distance o_l larger than the maximum theoretical distance $T_{l,\max}$ from v_{p_l} to a node inside the EPZ, i.e., the maximum of the matrix row $T_{l,*}$. We do this because these activities could never cover the shortest path towards the protected location. For each entry gate Y_j, our algorithm then discards outlier points where the reported distance deviates significantly from the distance for other points within the same entry gate. Concretely, we consider a deviation of more than three standard deviations from the mean significant.

4.3.4 Predict Protected Location. As the points that constitute one entry gate do not overlap exactly, we use least absolute deviation (LAD) regression to predict the most likely protected location across all these diffuse entry gates. For each node v_l inside the EPZ (i.e., each possible protected location), the algorithm calculates the sum of absolute differences across all points $v_{p_l} \in P'$ between the observed distance o_l and the theoretical shortest path distance $T_{v_{p_l},v_l}$ between v_{p_l} and v_l. The final predicted protected location is then the node $v_l \in V$ where this sum is minimal. Note that we retain Assumption A2 here, as the predicted location will always be located on the road graph. However, users may start or finish their activity away from the road, e.g., on their private grounds. We offset this violation of our assumption through the definition of an error threshold τ_e, below which the protected location is sufficiently closely predicted to deanonymize the user. In Appendix A, we empirically determine an acceptable error threshold based on real-world activities.

5 DATA COLLECTION

A large pool of real user data from a fitness tracking social network is required to further evaluate the plausibility of our attack against state-of-the-art EPZ implementations. For this, we use Strava because it is one of the most popular fitness tracking social networks, with over four billion total activities recorded so far [49]. In this section, we explain the methodology we used to collect user data from Strava and further analyze this data set to obtain useful insights into the privacy habits of users and fitness trackers.

5.1 Methodology

The scope of our data collection is a period of one week starting from 11 July 2021. By inspecting timestamps and elapsed time of activities, we conclude that IDs of activities are assigned sequentially rather than randomly upon uploading. However, due to delays in uploading activities to Strava, the temporal order is not sequential. We identify the first activity ID globally assigned after 11 July 2021 00:00 UTC. We then visit each next 9,000th public activity from approximately 36 million activities uploaded during this week. If this activity is not public, does not exist anymore, or was completed before 11 July but only uploaded after, we consider the next sequential activity. For this (public) activity, we identify the user who created it. This ultimately provides us with a randomly generated, representative sample of 4,000 users.

Table 2: Empirically selected parameters for Algorithms 1 and 2, and the success rate metric, as used in our evaluation.

(a) **Algorithm 1**

Parameter	Value
$\tau_{converged}$	10 m
$\tau_{disjoint}$	1600 m

(b) **Algorithm 2**

Param.	Value	Param.	Value
d_{chain}	3 m	$minPts$	1
τ_{snap}	10 m	ϵ	20 m

(c) **Success rate**

Param.	Value
τ_e	22.95 m

For each user in our sample, we retrieve user information (i.e., nationality) and the IDs of their public activities with map data. For this, we scrape and parse the *overview* section of the athlete's profile page[4] using Selenium [47]. Then, for each activity (ID), we extract its total distance and type from the `strava.com/activities/ID` page. We also collect the elevation profile including the GPS track points as coordinate pairs with the corresponding elevation and accumulated distance data from the `strava.com/stream/ID` API endpoint. Since an authenticated user can make a maximum of 375 requests to Strava's stream API per day, we use multiple accounts to speed up the download process. Despite downloading with multiple accounts to circumvent rate limiting, our data set took three months to collect. Before storing our collected data in our database, we pseudonymized the data by replacing original user IDs with an autoincremented primary key upon request of our IRB. We repeat the same procedure for activity IDs.

To obtain the road graph for each EPZ, we use the OSMnx framework [8] to download the OpenStreetMap road network in a graph format (see Figure 4a).

5.2 Data Characteristics

With the aforementioned methodology, we collected a data set of 1,404,886 activities created by 4,000 users. We plot several distributions within this data set in Appendix B. The distribution of number of activities per user is shown in Figure 11, with a median of 136 activities per user. Our data set is geographically diverse as shown in Figure 12, containing activity endpoints from 160 different countries. The data set provides different densities and layouts of road networks, which is an important factor for the efficacy of our attack as shown in Section 6.4.4.

We observe that 461 users (11.53% of our total data set) use at least one EPZ to cloak activities. Figure 13 shows the distribution of EPZ radii for these users, as determined through Algorithm 1 (with parameters from Table 2a). Radii up to 400m have a 65.50% share, indicating that smaller radii are more popular than larger ones. 35.08% of protected activities in our data have only one cloaked endpoint, making them usable for the total distance scenario. We observe non-fixed EPZ radii in our data set that have a statistically insignificant distortion score and fit the endpoints well through visual inspection. We assume that user-selected, non-fixed radii are a legacy feature that is not available anymore.

[4] `strava.com/athletes/ATHLETE_ID`

6 EVALUATION

6.1 Ground Truth Definition

To assess the effectiveness of our algorithm, we require a ground truth of known protected locations that we can compare to the predictions made by our algorithm. To do so, we use the 1,312,250 uncloaked[5] activities generated by the 4,000 users in our data set (Section 5). For each user, we search their activities that are part of clusters of at least 15 endpoints that fall within 50 meters of one another. Since most fitness tracking social networks use an autocomplete address form for the creation of privacy zones, we then designate the streetside location of the closest address to these cluster's centroids as our ground-truth location $v_{GT} \in V$ (i.e., the point that a user would configure as a protected location). We believe that inferring likely protected locations from uncloaked activities is the most feasible way to generate a sufficiently large-scale yet reliable ground truth. We note that our approach to obtaining ground truth is almost identical to that of the existing state of the art [24].

Using the aforementioned methodology, we constructed 4,689 ground-truth locations for 2,527 users (63.18 % of entire data set). We were unable to construct at least one reliable ground-truth location for the other 1,473 users due to a lack of geographically concentrated, uncloaked activities of the *Walk*, *Run* or *Ride* type.

6.2 Prediction Evaluation

For each ground-truth location, we synthesize an EPZ, with its center randomly translated from that location, for each radius in the set of radii available in Strava. We consider this set of radii as exemplar since Strava is one of the most popular FTSNs, and since it supports the largest EPZ radii. In the case that multiple protected locations of the same user are located inside the same EPZ, i.e., closer to each other than the EPZ radius (8% of 200m EPZs), we only synthesize an EPZ around the protected location with the most activity observations. We then simulate the cloaking of these –previously uncloaked– activities by removing all points that fall within the synthetic EPZ. Finally, we evaluate our attack by predicting the corresponding protected location of each (synthesized) EPZ using Algorithm 2 with the locally cloaked activity data as input. We then compare this prediction (i.e., the result of our algorithm) with the ground-truth location.

We measure the predictive error of Algorithm 2 by constructing confidence intervals (CIs) using bootstrapping [18], a random sampling method with replacement. We run our algorithm (using the parameters in Table 2b) 1000 times with resampled data from the observed activities a, retaining the original number of cloaked activities for this EPZ. This yields 1000 (not necessarily distinct) predicted locations $V_{pred} \subseteq V$, which we denote as the CI constructed by bootstrapping. This finally allows us to estimate the probability distribution $\widehat{Pr}(v|a)$, i.e., protected locations $v \in V$ given the user's activities a, with the probability quantified as the number of times the location was predicted out of the 1000 runs of our algorithm. Note that this probability is, therefore, zero for locations that were never predicted, i.e., $v_p \in V_{pred} \Leftrightarrow \widehat{Pr}(v_p|a) > 0$. We compute an extended CI $V_{pred,ext} \subseteq V$ to account for the 'overshoots' caused by activities starting or finishing away from the road graph (Section 4.3.4). This extended CI encompasses the nodes $v_e \in V$ that lie within the error threshold τ_e of the predicted locations V_{pred}, i.e., $v_e \in V_{pred,ext} \Leftrightarrow \exists v_p \in V_{pred} : \text{dist}(v_e, v_p) \leq \tau_e$. We further discuss the overshoots, and empirically determine τ_e in Appendix A.

We subject our predictions to the following privacy metrics, which except for success rate, we compute separately for each EPZ:

Success rate [36] is defined as the percentage of EPZs for which the attacker is 'successful' (binary value). An attacker is considered succesful when the ground-truth location lays inside the extended CI $V_{pred,ext}$ or, in other words, the predicted protected location is sufficiently close to the ground-truth location to deanonymize the user.

$$v_{GT} \stackrel{?}{\in} V_{pred,ext} \tag{1}$$

Correctness [46] is quantified as the sum of Euclidean distances between the true outcome (ground-truth location) v_{GT} and each node $v \in V$, weighted by the probability distribution $\widehat{Pr}(v|a)$ (non-zero only if the location was predicted).

$$\sum_{v \in V} \widehat{Pr}(v|a)\, \text{dist}(v, v_{GT}) \tag{2}$$

Accuracy [46] is quantified as the width of the confidence interval constructed by bootstrapping (i.e., the number of unique predicted locations v_{pred}). Note that a higher value for accuracy reflects a wider confidence interval, so the adversary is less confident of their prediction, and privacy improves.

$$|V_{pred}| \tag{3}$$

Reduction of the k-anonymity set refers to the k-anonymity set generated by an EPZ covering a ground-truth location, with k the number of nodes of the chained road graph inside the EPZ (i.e., all possible protected locations). The reduction is then defined as the proportion of k minus the number of nodes inside the extended CI, over k.

$$\frac{k - |V_{pred,ext}|}{k} \tag{4}$$

Size of Uncertainty Region [13] is defined as the area of the union of (possibly disjoint) circles around the predicted nodes in the confidence interval with a radius equal to the chaining distance.

$$Area\Big(\bigcup_{v_p \in V_{pred}} C_{v_p, d_{chain}} \Big) \tag{5}$$

Certainty [46] is the Shannon entropy [44] of the estimate distribution $\widehat{Pr}(v|a)$ and represents how concentrated the probability distribution is, but lacks a notion of (spatial) neighborhoods. A higher entropy value indicates a less certain adversary.

$$-\sum_{v \in V} \widehat{Pr}(v|a) \log\Big(\widehat{Pr}(v|a)\Big) \tag{6}$$

Spatial Certainty is based on Karlstrom and Ceccato's entropy [26]. Instead of using the probability of a single node v, we use the logarithm of the neighborhood probability $\widehat{Pr_n}(v|a)$ as the surprisal term of the entropy formula. Therefore, this represents how *spatially* concentrated the distribution is. We refer the reader to Section 11 for more details on this new metric.

$$-\sum_{v \in V} \widehat{Pr}(v|a) \log\Big(\widehat{Pr_n}(v|a)\Big) \tag{7}$$

[5]While FTSNs do not explicitly show that an EPZ is used, we infer whether an activity is cloaked by checking if there is a discrepancy in the visible distance and total distance of an activity.

Table 3: Susceptibility to our attack of popular FTSNs' EPZ implementations, for the two scenarios from Section 4.3.

Application	Inner	Total
Strava [48]	✓	✓
Garmin Connect [21]	✗	✓
Relive [42]	✗	✗
Komoot [27]	✗	✓
Map My Tracks [31]	✓	✓
Ride With GPS [43]	✓	✓

Degree of Anonymity [15] is the entropy of the estimate distribution $\widehat{Pr}(v|a)$, normalized by the maximum value entropy $H_0(V)$ when all nodes inside the EPZ are equally likely.

$$\frac{-\sum_{v \in V} \widehat{Pr}(v|a) \log_2 \left(\widehat{Pr}(v|a)\right)}{H_0(V)} \tag{8}$$

6.3 Results

In this subsection, we present the results of our attack against actual fitness tracking social networks. First, we analyze whether FTSNs leak sufficient distance metadata through their APIs to enable our attack. Table 3 shows that the EPZ implementations of 3 and 5 fitness tracking social networks are vulnerable to our attacks in an inner and total distance scenario (Section 4.3), respectively. Relive is the only network that is never vulnerable to our attack, since it enforces EPZs by truncating activities to only the track outside the EPZ when uploading. However, this truncation significantly reduces usability (Section 7).

Next, we examine the efficacy of our attack with the privacy metrics from Section 6.2 on the collected data set from Section 5 against all radii in the studied radii set. The results for the inner distance and total distance scenario are shown in Tables 4 and 5 (Appendix C) respectively, and visualized altogether in Figure 6. Note that, except for success rate, which is computed as a percentage of binary success across all EPZs, we report the median across all EPZs of the per-EPZ privacy metric values.

In the inner distance scenario, we achieve a success rate of up to 85% for EPZs with a radius of 200 m, the most popular option across Strava users (Figure 13). We notice a decrease in efficacy of the attack as the EPZ radius increases, as depicted in Figure 6. Nevertheless, even for the largest radius (1600 m), we still successfully deanonymize 39% of protected locations. We achieve almost identical success rates for the total distance scenario.

For larger radii, the number of nodes inside the EPZ (i.e., the candidate protected locations), as well as the number of nodes at roughly the same distance from the entry gates of the EPZ increases. This leads to more *confusion* between candidate locations in our LAD regression from Section 4.3.4, since more candidate nodes have similar distances. Increasing the radius, therefore, yields predictions with increased accuracy (i.e., larger confidence intervals), which in turn, enlarges the uncertainty region from the adversary's perspective. Reduction of the k-anonymity set, on the other hand, increases because the growth of the accuracy is smaller than the growth of the number of candidate nodes. The median accuracy amounts to 10 nodes for the inner distance scenario and a 200 m

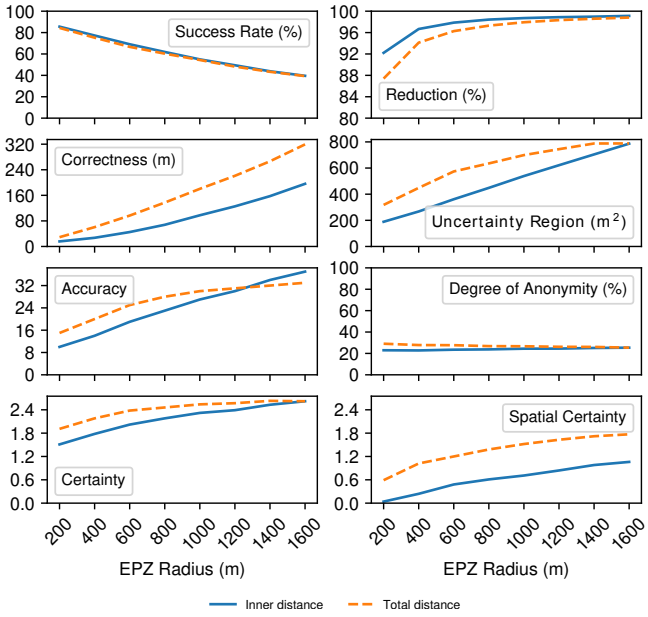


Figure 6: Privacy metrics from Section 6.2 for the predictions resulting from our attack, representing the attack's efficacy.

radius, resulting in a 92% reduction of the k-anonymity set and an uncertainty area of $188m^2$. In the total distance scenario, we have fewer 'suitable' activity endpoints compared to the inner distance scenario: as mentioned in Section 4.3.3, we filter out the activity endpoints where the user did not follow the shortest path. This reduction in suitable endpoints negatively impacts the performance of the attack, as we will discuss in Section 6.4. For smaller radii, the total distance attack has a slightly higher accuracy, resulting in smaller reduction and larger uncertainty areas compared to the inner distance attack.

The confusion also negatively impacts the correctness of our prediction, since the probability of predicting nodes other than the ground truth increases. Moreover, the larger the radius, the less probable it is that a user takes the shortest path inside the EPZ, violating Assumption A4. To a lesser extent, the number of activities our model can use for its prediction decreases as larger radii are (nearly) enveloping entire activities. We achieve a median correctness of 15m for the inner distance scenario and a 200 m radius vs. 29m in the total distance scenario.

An adversary will have similar certainty for both distance scenarios, steadily increasing with increasing radius (which also increases the number of nodes), meaning that an attacker is less confident in selecting one solution $v_p \in V_{pred}$. However, using spatial certainty as a metric, an attacker is more confident in (geographically) pinpointing one location in the inner distance scenario than the total distance scenario. For the most popular radius, our attack achieves a median spatial certainty of almost 0, meaning that we are able to pinpoint a single location. Finally, the Degree of Anonymity remains almost constant with increasing radius for both attack scenarios, since the certainty of the attacker increases linearly with the logarithm of the k-anonymity set size.

Comparison With Prior Work. Our attack achieves a success rate comparable to that of Hassan et al.'s main attack which uses circle fitting [24]. Hassan et al. report a global 84% success rate, and while their results are not fully broken down by radius, they report a 44% success rate for 1 km radii. We achieve a success rate of up to 85% (for 200 m radii) and a 55% success rate for 1 km radii. However, our attack works for a harder-to-break EPZ implementation that includes spatial cloaking, for which Hassan et al.'s main attack does not work, and we use a lower error threshold (τ_e = 22.95 m for us vs. 50 m for Hassan et al.). In addition, Hassan et al. proposed spatial cloaking as a countermeasure, and evaluated an alternative interpolation attack on EPZs to which spatial cloaking has been applied [24, Section 6.4]. Their attack relies only on route direction, resulting in a success rate of at most 45% (for a radius of 200 m). Our attack using regression across distance metadata performs up to two times better (for similar radii) than Hassan et al.'s alternative attack, with our attack having up to 85% success, again at a stricter error threshold. Therefore, we effectively circumvent the countermeasure of Hassan et al. adopted by major FTSNs.

Mink et al. [35] found that human users can infer up to 68% of all sensitive locations in their data set, but again for the original EPZ implementation without spatial cloaking and with a laxer error threshold. In comparison, our (automated) attack achieves a higher success rate and works on EPZs with spatial cloaking by leveraging topological information. While Mink et al. propose that humans may also use topological information for visual inference, we conjecture that the need to visually trace the precise distances travelled within the EPZ will make the visual inference task much less successful at deanonymizing spatially cloaked protected locations than our LAD regression.

6.4 Sensitivity Analysis

In order to better understand the conditions in which our attack performs better or worse, and eventually develop more effective countermeasures, we now analyze attack performance when different factors are varied.

6.4.1 Suitable Activity Endpoints. We first analyze whether more activities and endpoints lead to better performance. As the number of activity endpoints (i.e., observations for LAD regression in Section 4.3.4) increases, the variance of the difference between observed and theoretical distances for our prediction will decrease. More activities contribute to a slightly higher success rate for predictions, independent of the EPZ radius, as Figure 7 indicates.

6.4.2 Entry Gates. An increased number of entry gates yields a slightly higher success rate for smaller radii, as can be seen in Figure 7. The effect for larger radii is nullified by users not taking the shortest path. In fact, the geographic distribution of entry gates has a higher impact on the success rate of our predictions than the number of entry gates.

6.4.3 Blind Spot Angle. We present the *maximum blind spot angle* as a metric to measure this geographic distribution and define it as the maximum angle between entry gates relative to the center of the EPZ. Smaller blind spot angles provide more diversified observations for the LAD regression in Section 4.3.4 than observations within one entry gate. These diverse observations increase the difference between theoretical shortest path distances and observed distances of erroneous nodes, making it less likely that the LAD regression selects the wrong locations. This explains the higher success rate, as can be seen in Figure 7.

6.4.4 Density of Road Network. As the EPZ radius increases, we observe an increasing negative effect of increased street density (expressed as meters of road per square kilometer) on the success rate of our predictions, as Figure 7 shows. As the density increases, so does the number of nodes with the same distance from the entry gates of the EPZ. This increased number of candidate nodes causes confusion in the LAD regression, resulting in larger confidence intervals and the prediction of incorrect nodes which, in turn, has a negative influence on the success rate. Moreover, a denser road network gives the user more routes to take that are not the shortest path, again violating Assumption A4.

As one might expect for a distance-based attack, performance primarily depends on geographic factors. Increased geographic diversity in entry gates reduces confusion between candidate locations, particularly with sparser street grids and with smaller maximum blind spots. An increased number of activities is, therefore, only useful for our predictions if they introduce additional geographic disparity. Otherwise, their effect on the performance of our attack is moderate.

7 COUNTERMEASURES

Our evaluation shows that identifying protected locations remains feasible with current EPZ implementations. While other privacy defenses exist, users tend to use these in combination with EPZs, and users still find EPZs efficient [35]. In this section, we therefore develop and evaluate potential countermeasures that support the continued use of EPZs by making EPZs more resilient against our attack. We evaluate our countermeasures on the inner distance scenario of an EPZ with a radius of 400 meters. This inner distance scenario on a small radius is favorable from the adversary's viewpoint, yet accounts for the majority of EPZs in our data set as discussed in Section 5.2. We also discuss the usability and privacy implications and trade-offs of our countermeasures.

7.1 Distance-Focused Countermeasures

Our attack primarily relies on the availability of the distance covered within the EPZ. Countermeasures could seek to obfuscate these distances, in order to increase the error in our regression. This also has the effect of altering the total traveled distance.

C1 – Generalization. By reducing the precision of any reported distance shown to a non-owner user [51], the adversary would be unable to reliably determine the distance between the actual and cloaked start/finish points (whether inner or total distances). This reduces the precision of the last step of Algorithm 2 (Section 4.3.4), in which we search the point where the theoretical shortest path distances best correspond to the actually observed distances. We implement generalization by rounding distances to the nearest multiple of a certain integer value. In Figure 8, we evaluate the performance of our attack given different roundings of the inner

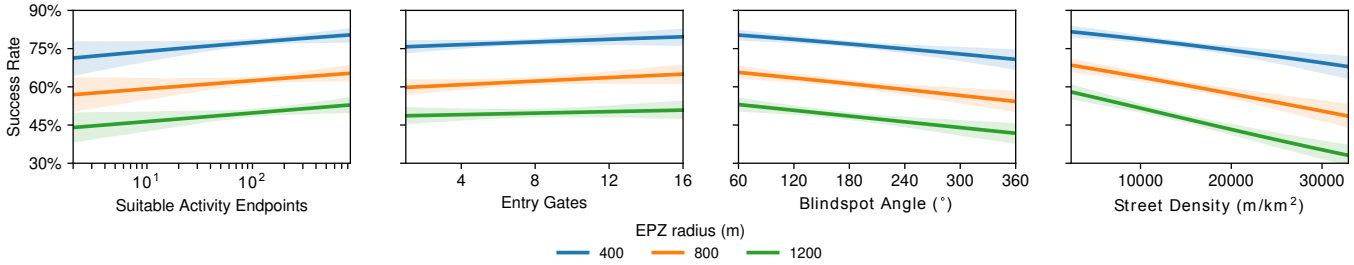


Figure 7: Logistic regression of the independent variable 'success' of our attack for the predictor variables selected in our sensitivity analysis, across three EPZ radii.

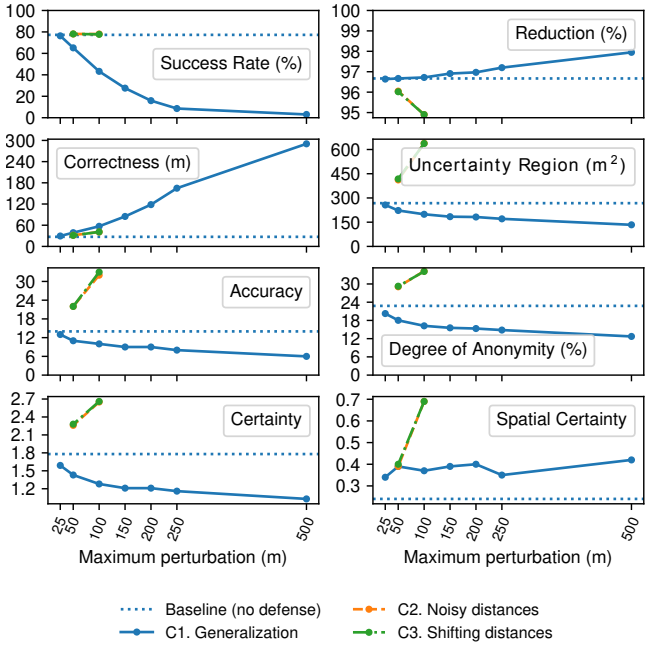


Figure 8: Impact of countermeasures C1-C3[6] on the privacy metrics for our predictions and therefore on the efficacy of our attack, for the inner distance scenario on an EPZ with a 400 m radius. Full numerical results are presented in Table 6 (Appendix C).

distance; note that the maximum perturbation is half of the rounding distance. We find that the success rate fully breaks down after a rounding of around 500 meters, although even small roundings already largely reduce this success rate. Argyros et al. [6] similarly found that attack performance is inversely proportional to the magnitude of the applied rounding. As the rounding distance increases, the uncertainty of the adversary (wrongly) decreases: multiple observations get mapped onto the same generalized value leading to less confusion in the last step of Algorithm 2. This makes the adversary more certain of their false predictions.

C2 – Noisy distances. This countermeasure applies random noise to the reported distance, instead of rounding it. This adds more uncertainty to the predictions of the adversary, resulting in bigger confidence intervals and uncertainty regions, as depicted in Figure 8. However, multiple activities from the same entry gate would result in these random shifts being averaged out, causing the ground truth to be present in the confidence interval. With the ground truth still present in the confidence interval, we observe no change in the success rate.

C3 – Shifting distances. The reported endpoints are shifted by a fixed or random distance, while retaining the originally traveled distance as the total distance. However, similar to the noisy distances countermeasure, these random shifts may be averaged out across activities. In Figure 8, we see an increased uncertainty region of the adversary compared to the baseline. However, since the success rate has not changed, k-anonymity has not been restored.

C4 – Truncation. A more invasive countermeasure consists of eliminating the track portions lying within the EPZ entirely, by not including them in the reported total and accumulated distances, or even hiding the full track. This would effectively thwart our attack, as we can no longer infer where on the street grid inside the EPZ the activity may have started and ended; only a random guess among all possible protected locations remains possible.

The main disadvantage of altering the reported distances lies in their negative usability impact. Whereas in other location-based services such as check-in apps, an error of several hundreds of meters may be acceptable [6], this may be less the case for fitness social networks. Part of the attraction of these networks comes from the gamification of exercise activities [7], such as achievements for covering certain distance goals or being able to compare across small performance differences (in the order of seconds) [14], which require high detail in activity data. Aggressive rounding of distances would result in losing the desired precision at which the distance and pace are measured, possibly leading to overestimated achievements. Removing the track portions inside the EPZ and shortening the track distance is, therefore, also unattractive: it would result in underestimating the achieved distance.

7.2 EPZ-Focused Countermeasures

Countermeasures could target the EPZ to decrease (the utility of) available data or the identification of the EPZ (Algorithm 1).

C5 – Increasing EPZ radii. An obvious countermeasure is increasing the EPZ radius. Our evaluation in Section 6.3 confirms that, for circular EPZs, the attack performance decreases when the EPZ radius increases. However, this severely reduces usability, particularly for shorter activities, as they may be entirely covered by the

[6]Note that metrics for C2 and C3 strongly overlap.

EPZ. This might also be why larger EPZ radii are less popular than smaller radii (as seen in our data set in Section 5.2).

C6 – Complex EPZs/cloaking. Beyond circular EPZs with a small set of possible radii, different EPZ layouts could be implemented: circles with any radius or even any shape. Hassan et al. [24] proposed a similar strategy by hiding a (small) track portion of random[7] length beyond the EPZ border ("Fuzz EPZ Intersection Points"). Alternatively, the concept of a 'zone' could be discarded, and instead a fixed or variable distance from the start and finish points could be cloaked, as Strava has recently implemented [32]. For our attack, this only affects the phase of identifying EPZs, as this step will be less reliable due to the noise applied to the cloaked endpoints. Afterwards, the countermeasure has no effect on our location prediction algorithm: the inner/total distances and endpoints remain available, and allow for regression.

More complex zones require more complex identification algorithms, beyond the least squares fit for circular EPZs (Algorithm 1). These algorithms may be less accurate at identifying EPZs. More complex zones may also make the removal of deviating activities (Section 4.3.3) less effective, as the threshold for becoming an outlier becomes less likely to be met. However, as stated in Section 4.2, our attack still works with imprecise EPZ identification. Countermeasures that make EPZ discovery harder, therefore, cannot fully thwart our attack. Crucially, these countermeasures do not affect the availability nor accuracy of the reported distances. Any increase in the search space due to a less compact EPZ will reduce the attack performance by increasing the likelihood that an erroneous location is predicted, and it will make the attack more computationally expensive.

Finally, a number of apparent countermeasures may seem effective at first, but can potentially *improve* the efficacy of our attack. Regeneration of an EPZ recomputes the endpoints for every (future) activity, and may, therefore, generate additional entry gates, as the edge of the EPZ has now shifted. This yields additional and more diverse data, which could improve the correctness of our attack, since our evaluation in Section 6.4 shows that correctness tends to decrease when there are more entry gates and activities. Next, smoothing the track by map matching nodes to the road network would remove any (small) deviations that cause the actual traveled distance to not match the theoretical shortest path exactly, and would, therefore, make the LAD regression step more accurate.

In summary, distance-based countermeasures and in particular generalization are the most effective, but can severely reduce usability. Countermeasures that target EPZ discovery are less invasive, but only partially prevent our attack.

8 RELATED WORK

Two recent works have analyzed vulnerabilities in previous implementations of EPZs where no spatial cloaking was applied and the center of the EPZ was therefore the protected location. In 2018, Hassan et al. [24] were able to infer EPZs and their protected locations by fitting circles between pairs of endpoints. They identified 84% of 432,022 athletes across 2.3 million EPZ-enabled Strava activities. In 2022, Mink et al. [35] showed that users could visually identify up to 68% of protected locations when asked to draw the EPZ between activity endpoints overlaid on a map and pinpoint the protected location. Hassan et al. [24] proposed several countermeasures, which were implemented by some fitness tracking networks. Crucially, both works therefore only prove the vulnerability of an EPZ implementation that is by now arguably outdated. In contrast to both works, our attack breaks the current state-of-the-art EPZs, i.e., those when spatial cloaking is present, with a comparable or higher success rate than the prior work, on a harder-to-break EPZ implementation. We also analyze in depth to which factors of privacy zones our attack is sensitive.

Other work has explored other privacy concerns in sharing location data on fitness tracking social networks. Beyond privacy zones, Meteriz et al. [34] found that elevation profiles could be sufficient to recover a location at borough- or city-level, even if the location data is not shared. They require prior knowledge of potentially visited locations, which are predicted at very low granularity, unlike our attack. Alqhatani and Lipfore [2], Zimmer et al. [56], Gabriele and Chiasson [20], Couture [14], and Mink et al. [35] described how users are somewhat aware of the privacy implications of sharing location data on fitness tracking social networks, but that this awareness may be insufficient. They also found that users differ in their sensitivity to having sensitive locations publicly available, correlating with concerns on personal space and physical safety. Mink et al. [35] found that users consider EPZs an effective privacy mechanism, but that these users would mostly use EPZs together with other privacy mechanisms.

Beyond fitness services, prior work has evaluated the feasibility of de-anonymizing implementations for location proximity, where the distance to nearby users is shown instead of their actual location. Li et al. [29] developed attacks for three popular location proximity services, accurate to up to 25 meters. Argyros et al. [40, 6] showed that major location proximity services remained vulnerable to location inference attacks, despite existing countermeasures. These attacks were sufficiently performant to enable real-time tracking. These two studies proposed some form of spatial cloaking [23] as an effective countermeasure. Qin et al. [41] and Zhao et al. [55] showed for 4 and 29 apps respectively that revealed distances enable trilateration attacks.

On the side of defenses, Gruteser and Grunwald [23] proposed spatial and temporal cloaking, where an error is introduced to the location information, e.g., by decreasing resolution or applying random noise. Cheng et al. [13] evaluated the trade-off between such cloaking and the quality of the provided service, and propose imprecise queries across cloaked locations to improve this trade-off. Ardagna et al. [5] concretely defined obfuscation techniques on circular zones (compare to circular EPZs). Duckham and Kulik [17] formalized obfuscation as a means to achieve location privacy, implemented through precision reduction (similar to cloaking). Andrés et al. [3] formally introduced geo-indistinguishability, where a user reveals a sufficiently approximate location to receive a desired service, instead of their exact location. In this and follow-up work [9, 11, 12], they proposed several techniques for achieving this property.

[7] A fixed shift would be equivalent to increasing the EPZ radius.

9 ETHICS AND RESPONSIBLE DISCLOSURE

We disclosed this research project to our university's privacy and ethics board before we collected any data or ran any experiments. Our project was formally approved, and we implemented all recommendations regarding pseudonymization of user and activity IDs, and data retention on secure internal network storage servers.

During the course of this project, we discovered several fundamental problems in how fitness tracking social networks implement EPZs. We disclosed our findings to the affected parties from Table 3 by sending them a draft of this manuscript and by formulating recommendations for improving the privacy of their users.

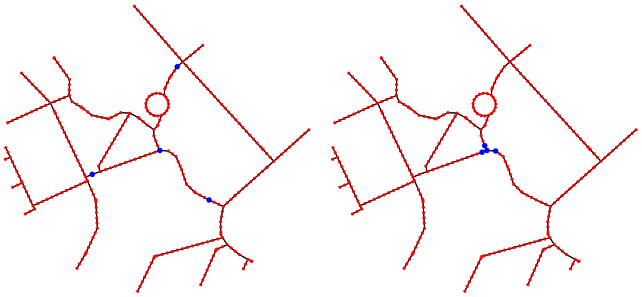
10 CONCLUSION

Major fitness tracking social networks have introduced endpoint privacy zones as a tool to protect sensitive locations from being revealed to malicious actors. However, despite the usage of spatial cloaking, we show that these protected locations can still be discovered reliably. Our attack leverages the reported distances traveled within the EPZ, as well as the layout of the street grid to deanonymize protected locations with a success rate of up to 85%. While distance-based countermeasures such as generalization can be effective at thwarting our attack, they can also severely reduce usability. Networks must, therefore, carefully consider which functionality they provide while guaranteeing user privacy.

11 ADDENDUM: SPATIAL CERTAINTY

We design a new metric to measure *spatial certainty*, which represents the geographical closeness of predictions. For example, consider two situations (a) and (b), depicted in Figure 9, where an attacker predicts four locations inside an EPZ with uniform probability. Whereas the predictions in (a) are geographically dispersed, the predictions in (b) are geographically concentrated.

Shokri [46] defined certainty as the Shannon entropy [44] of the estimate probability distribution $\widehat{Pr}(v|a)$. This entropy shows how concentrated $\widehat{Pr}(v|a)$ is and, thus, how easy it is to pinpoint a single outcome $v_p \in V_{pred}$. Since this distribution $\widehat{Pr}(v|a)$ is the same in (a) as in (b), it will result in the same value for certainty. However, the certainty value does not give a notion of the spatial concentration of predictions.



(a) Geographically dispersed predictions of an adversary. (certainty=1.39, spatial certainty=1.39)

(b) Geographically concentrated predictions of an adversary. (certainty=1.39, spatial certainty=0.0)

Figure 9: Geographical plot of predictions of an adversary. Each node $v \in V$, displayed in red, is a potential protected location. Each actual predicted node $v_p \in V_{pred}$ is displayed in blue.

By increasing the chaining resolution (i.e., reducing the chaining distance d_{chain}, as explained in Section 4.3), the number of possible protected nodes $v \in V$ increases. This will make it harder for an attacker to pinpoint a single solution $v_p \in V_{pred}$. However, we only claim to predict a location with τ_e precision. Therefore, we should consider neighborhoods of nodes within τ_e when determining if a prediction is sufficiently close to the protected location.

We additionally define spatial certainty as Karlstrom and Ceccato's entropy [26]. Instead of using the logarithm of the probability of a single node v as the surprisal term in the entropy formula, we use the logarithm of the sum of all neighboring nodes v_j probabilities of v including v itself:

$$-\sum_{v \in V} \widehat{Pr}(v|a) \log\left(\widehat{Pr_n}(v|a)\right)$$

A node v_j is considered a neighbor of v if the Euclidean distance between both nodes is less than or equal to τ_e:

$$\widehat{Pr_n}(v_i|a) = \sum_{v_j \in V} w_{ij} \widehat{Pr}(v_j|a) \quad \text{with} \quad w_{ij} = \begin{cases} 1, & \text{if } \mathrm{dist}(v_i, v_p) \leq \tau_e \\ 0, & \text{otherwise} \end{cases}$$

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their valuable and constructive feedback, as well as the Security Analytics SIG at DistriNet. This research is partially funded by the Research Fund KU Leuven, and by the Flemish Research Programme Cybersecurity. Victor Le Pochat holds a PhD Fellowship of the Research Foundation Flanders - FWO (11A3421N). Map tiles by Stamen Design, under CC BY 3.0. Map data from OpenStreetMap, under ODbL.

REFERENCES

[1] Adidas. 2021. adidas Runtastic: adidas Running & adidas Training apps. Retrieved Sept. 21, 2021 from https://www.runtastic.com/.

[2] Abdulmajeed Alqhatani and Heather Richter Lipford. 2019. "There is nothing that I need to keep secret": Sharing Practices and Concerns of Wearable Fitness Data. In *15th Symposium on Usable Privacy and Security* (SOUPS '19), 421–434.

[3] Miguel E. Andrés, Nicolás E. Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. 2013. Geo-Indistinguishability: Differential Privacy for Location-Based Systems. In *2013 ACM SIGSAC Conference on Computer and Communications Security* (CCS '13), 901–914. DOI: 10.1145/2508859.2516735.

[4] Carmen Ang. 2020. The Growth of Home Fitness Apps. Visual Capitalist. (Sept. 10, 2020). https://www.visualcapitalist.com/the-growth-of-home-fitness-apps-2020/.

[5] C. A. Ardagna, M. Cremonini, E. Damiani, S. De Capitani di Vimercati, and P. Samarati. 2007. Location Privacy Protection Through Obfuscation-Based Techniques. In *IFIP Annual Conference on Data and Applications Security and Privacy*, 47–60. DOI: 10.1007/978-3-540-73538-0_4.

[6] George Argyros, Theofilos Petsios, Suphannee Sivakorn, Angelos D. Keromytis, and Jason Polakis. 2017. Evaluating the Privacy Guarantees of Location Proximity Services. *ACM Transactions on Privacy and Security*, 19, 4, Article 12, (Feb. 2017), 31 pages. DOI: 10.1145/3007209.

[7] Paul Barratt. 2017. Healthy competition: A qualitative study investigating persuasive technologies and the gamification of cycling. *Health & Place*, 46, (July 2017), 328–336. DOI: 10.1016/j.healthplace.2016.09.009.

[8] Geoff Boeing. 2017. OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65, 126–139. DOI: 10.1016/j.compenvurbsys.2017.05.004.

[9] Nicolás E. Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. 2014. Optimal Geo-Indistinguishable Mechanisms for Location Privacy. In *2014 ACM SIGSAC Conference on Computer and Communications Security* (CCS '14), 251–262. DOI: 10.1145/2660267.2660345.

[10] Kaya Burgess. 2018. Thieves 'followed rider on his Strava app' to make off with £12,500 in bikes. *The Times*, (Sept. 22, 2018). https://www.thetimes.co.uk/article/thieves-followed-rider-on-his-strava-app-to-make-off-with-12-500-in-bikes-xqjrlgj0f.

[11] Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Marco Stronati. 2014. A Predictive Differentially-Private Mechanism for Mobility Traces. In *Privacy Enhancing Technologies*, 21–41. DOI: 10.1007/978-3-319-08506-7_2.

[12] Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Marco Stronati. 2015. Constructing elastic distinguishability metrics for location privacy. *Proceedings on Privacy Enhancing Technologies*, 2015, 2, 156–170. DOI: 10.1515/popets-2015-0023.

[13] Reynold Cheng, Yu Zhang, Elisa Bertino, and Sunil Prabhakar. 2006. Preserving User Location Privacy in Mobile Data Management Infrastructures. In *Privacy Enhancing Technologies*, 393–412. DOI: 10.1007/11957454_23.

[14] Jesse Couture. 2021. Reflections from the 'Strava-sphere': Kudos, community, and (self-)surveillance on a social network for athletes. *Qualitative Research in Sport, Exercise and Health*, 13, 1, 184–200. DOI: 10.1080/2159676X.2020.1836514.

[15] Claudia Díaz, Stefaan Seys, Joris Claessens, and Bart Preneel. 2003. Towards Measuring Anonymity. In *Privacy Enhancing Technologies*. Springer Berlin Heidelberg, 54–68. DOI: 10.1007/3-540-36467-6_5.

[16] E. W. Dijkstra. 1959. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, 1, 1, (Dec. 1959), 269–271. DOI: 10.1007/BF01386390.

[17] Matt Duckham and Lars Kulik. 2005. A Formal Model of Obfuscation and Negotiation for Location Privacy. In *International Conference on Pervasive Computing*, 152–170. DOI: 10.1007/11428572_10.

[18] B. Efron. 1979. Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, 7, 1, 1–26. DOI: 10.1214/aos/1176344552.

[19] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *2nd International Conference on Knowledge Discovery and Data Mining* (KDD'96). AAAI Press, 226–231.

[20] Sandra Gabriele and Sonia Chiasson. 2020. Understanding Fitness Tracker Users' Security and Privacy Knowledge, Attitudes and Behaviours. In *2020 CHI Conference on Human Factors in Computing Systems* (CHI '20). DOI: 10.1145/3313831.3376651.

[21] Garmin. 2021. Garmin Connect | Free Online Fitness Community. Retrieved Sept. 21, 2021 from https://connect.garmin.com/.

[22] Ralph Gross, Alessandro Acquisti, and H. John Heinz. 2005. Information revelation and privacy in online social networks. In *2005 ACM workshop on Privacy in the electronic society* (WPES '05). DOI: 10.1145/1102199.1102214.

[23] Marco Gruteser and Dirk Grunwald. 2003. Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. In *1st international conference on Mobile systems, applications and services* (MobiSys '03). DOI: 10.1145/1066116.1189037.

[24] Wajih Ul Hassan, Saad Hussain, and Adam Bates. 2018. Analysis of Privacy Protections in Fitness Tracking Social Networks -or- You can run, but can you hide? In *27th USENIX Security Symposium*, 497–512.

[25] Alex Hern. 2018. Fitness tracking app Strava gives away location of secret US army bases. *The Guardian*, (Jan. 28, 2018). https://www.theguardian.com/world/2018/jan/28/fitness-tracking-app-gives-away-location-of-secret-us-army-bases.

[26] Anders Karlström and Vânia Ceccato. 2002. A new information theoretical measure of global and local spatial association. *Jahrbuch fur Regionalwissenschaft*, 22, 1, 13–40.

[27] Komoot. 2021. Komoot | Find, plan and share your adventures. Retrieved Sept. 21, 2021 from https://www.komoot.com/.

[28] Komoot. 2020. Privacy Zones. (Aug. 18, 2020). https://support.komoot.com/hc/en-us/articles/360046595312.

[29] Muyuan Li, Haojin Zhu, Zhaoyu Gao, Si Chen, Le Yu, Shangqian Hu, and Kui Ren. 2014. All Your Location Are Belong to Us: Breaking Mobile Social Networks for Automated User Location Tracking. In *15th ACM International Symposium on Mobile Ad Hoc Networking and Computing* (MobiHoc '14), 43–52. DOI: 10.1145/2632951.2632953.

[30] J. MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *5th Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 1, 281–297.

[31] Map My Tracks. 2021. Map My Tracks - your active life in one app. Retrieved Sept. 21, 2021 from https://www.mapmytracks.com/.

[32] Meg. 2021. Edit Map Visibility. Strava Support. (Aug. 18, 2021). https://support.strava.com/hc/en-us/articles/115000173384.

[33] Meg. 2020. Your Privacy Defaults when you Create a Strava Account. Strava Support. (Oct. 13, 2020). https://support.strava.com/hc/en-us/articles/360034758331.

[34] Ülkü Meteriz, Necip Fazıl Yıldıran, Joongheon Kim, and David Mohaisen. 2020. Understanding the Potential Risks of Sharing Elevation Information on Fitness Applications. In *40th IEEE International Conference on Distributed Computing Systems* (ICDCS '20), 464–473. DOI: 10.1109/ICDCS47774.2020.00063.

[35] Jaron Mink, Amanda Rose Yuile, Uma Pal, Adam J. Aviv, and Adam Bates. 2022. Users Can Deduce Sensitive Locations Protected by Privacy Zones on Fitness Tracking Apps. In *2022 ACM CHI Conference on Human Factors in Computing Systems* (CHI '22). DOI: 10.1145/3491102.3502136.

[36] Arvind Narayanan and Vitaly Shmatikov. 2008. Robust De-anonymization of Large Sparse Datasets. In *2008 IEEE Symposium on Security and Privacy* (SP '08). DOI: 10.1109/sp.2008.33.

[37] Nick. 2019. Automatic privacy zone detection. Map My Tracks. (Jan. 28, 2019). https://www.mapmytracks.com/blog/entry/automatic-privacy-zone-detection.

[38] Nike. 2021. Nike Run Club App. Nike.com. Retrieved Sept. 21, 2021 from https://www.nike.com/nrc-app.

[39] Olivia Nuzzi. 2020. What It's Like to Get Doxed for Taking a Bike Ride. *Intelligencer*, (June 8, 2020). https://nymag.com/intelligencer/2020/06/what-its-like-to-get-doxed-for-taking-a-bike-ride.html.

[40] Iasonas Polakis, George Argyros, Theofilos Petsios, Suphannee Sivakorn, and Angelos D. Keromytis. 2015. Where's Wally?: Precise User Discovery Attacks in Location Proximity Services. In *22nd ACM SIGSAC Conference on Computer and Communications Security* (CCS '15). DOI: 10.1145/2810103.2813605.

[41] Guojun Qin, Constantinos Patsakis, and Mélanie Bouroche. 2014. Playing Hide and Seek with Mobile Dating Applications. In *29th IFIP International Information Security Conference*, 185–196. DOI: 10.1007/978-3-642-55415-5_15.

[42] Relive. 2021. Relive | Run, Ride, & more. Retrieved Sept. 21, 2021 from https://www.relive.cc/.

[43] Ride with GPS. 2021. Ride with GPS | Bike Route Planner and Cycling Navigation App. Retrieved Sept. 21, 2021 from https://ridewithgps.com/.

[44] C. E. Shannon. 1948. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27, 3, (July 1948), 379–423. DOI: 10.1002/j.1538-7305.1948.tb01338.x.

[45] Ax Sharma. 2020. Strava app shows your info to nearby users unless this setting is disabled. *BleepingComputer*, (Sept. 21, 2020). https://www.bleepingcomputer.com/news/security/strava-app-shows-your-info-to-nearby-users-unless-this-setting-is-disabled/.

[46] Reza Shokri, George Theodorakopoulos, Jean-Yves Le Boudec, and Jean-Pierre Hubaux. 2011. Quantifying Location Privacy. In *2011 IEEE Symposium on Security and Privacy* (SP '11). DOI: 10.1109/sp.2011.18.

[47] Software Freedom Conservancy. 2021. Selenium WebDriver. https://www.selenium.dev/documentation/webdriver/.

[48] Strava. 2021. Strava | Run and Cycling Tracking on the Social Network for Athletes. Retrieved Sept. 21, 2021 from https://www.strava.com.

[49] Strava. 2020. Strava releases 2020 Year In Sport Data Report. (Dec. 16, 2020). https://blog.strava.com/press/yis2020/.

[50] 2022. Strava's Global Community Continues Strong Growth Surpassing 100M Registered Athletes on the Platform. Strava. (May 24, 2022). https://blog.strava.com/press/100million/.

[51] Latanya Sweeney. 2002. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10, 05, (Oct. 2002), 571–588. DOI: 10.1142/s0218488502001650x.

[52] Emily Tseng, Rosanna Bellini, Nora McDonald, Matan Danos, Rachel Greenstadt, Damon McCoy, Nicola Dell, and Thomas Ristenpart. 2020. The Tools and Tactics Used in Intimate Partner Surveillance: An Analysis of Online Infidelity Forums. In *29th USENIX Security Symposium* (USENIX Security '20), 1893–1909.

[53] Under Armour. 2021. MapMyRun. Retrieved Sept. 21, 2021 from https://www.mapmyrun.com/.

[54] Delanie Woodlock. 2016. The Abuse of Technology in Domestic Violence and Stalking. *Violence Against Women*, 23, 5, (July 2016), 584–602. DOI: 10.1177/1077801216646277.

[55] Fanghua Zhao, Linan Gao, Yang Zhang, Zeyu Wang, Bo Wang, and Shanqing Guo. 2018. You Are Where You App: An Assessment on Location Privacy of Social Applications. In *2018 IEEE 29th International Symposium on Software Reliability Engineering* (ISSRE '18), 236–247. DOI: 10.1109/ISSRE.2018.00033.

[56] Michael Zimmer, Priya Kumar, Jessica Vitak, Yuting Liao, and Katie Chamberlain Kritikos. 2020. 'There's nothing really they can do with this information': unpacking how users manage privacy boundaries for personal fitness information. *Information, Communication & Society*, 23, 7, 1020–1037. DOI: 10.1080/1369118X.2018.1543442.

A ABSOLUTE MAP MATCHING ERROR

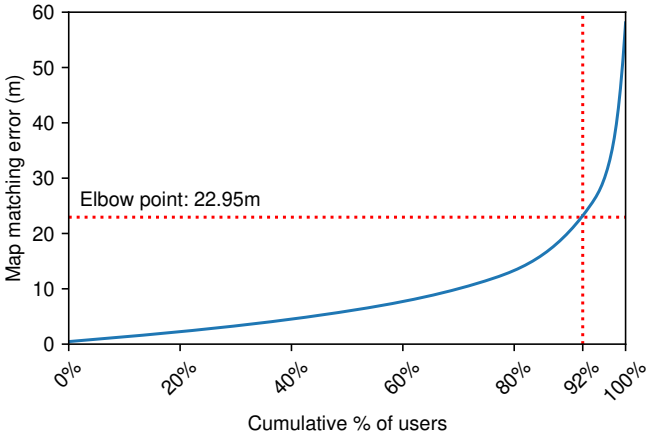


Figure 10: Distribution of the absolute map matching error across our 1.4 million Strava activities. We select the elbow point as our threshold for successful prediction.

A user may start and/or finish at a point away from the street grid (e.g., on private property), meaning that the uncloaked endpoints may be far away from our streetside ground-truth location. We, therefore, measure the absolute map matching error (i.e., the distance between the ground truth and the centroid of uncloaked endpoints that is used to construct the ground truth) and show its distribution in Figure 10. This additional distance leads to less precise predictions from our model, with the possibility of predicting 'overshoot' locations on the road network but away from our ground truth. In order to cover these corner cases, we define our prediction to be 'successful' (binary value) if the error of its location to the ground truth is less than an error threshold τ_e. We empirically select τ_e as the distance of the elbow point (22.95m) of the absolute map matching error distribution. 92% of users in our data set have a map matching error smaller than this distance.

B DATA SET CHARACTERISTICS

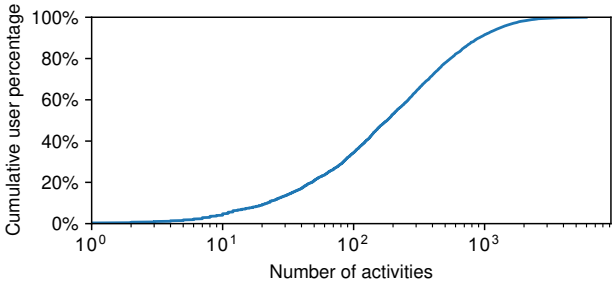


Figure 11: Cumulative distribution of number of activities per user over our data set of 4,000 Strava users.

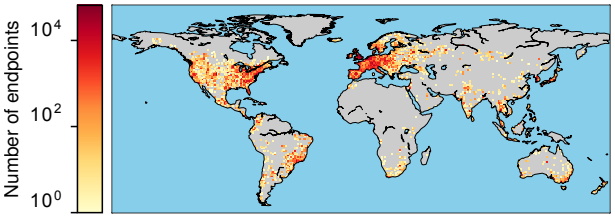


Figure 12: Geographic distribution of activity endpoints over our data set of 4,000 Strava users.

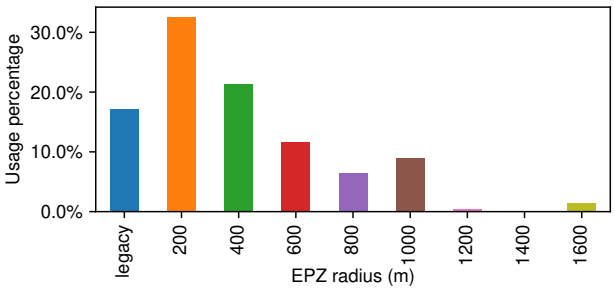


Figure 13: Distribution of selected EPZ radii across 461 Strava users that use the EPZ cloaking mechanism.

C FULL PRIVACY METRICS RESULTS

Table 4: Inner distance attack privacy metrics.

Radius (m)	Success Rate (%)	Correctness (m)	Accuracy	Reduction (%)	Uncertainty Region (m^2)	Certainty	Spatial Certainty	Degree of Anonymity (%)
200	85.55	15.79	10	92.21	188.97	1.51	0.04	22.96
400	77.26	27.11	14	96.67	267.75	1.78	0.24	22.79
600	69.09	45.09	19	97.86	361.25	2.02	0.48	23.49
800	61.89	67.83	23	98.43	448.34	2.18	0.61	23.76
1000	54.95	97.56	27	98.71	538.75	2.32	0.71	24.42
1200	49.44	125.49	30	98.88	621.66	2.39	0.84	24.49
1400	43.83	157.49	34	98.98	704.24	2.53	0.98	25.09
1600	39.58	196.03	37	99.12	786.48	2.62	1.06	25.36

Table 5: Total distance attack privacy metrics.

Radius (m)	Success Rate (%)	Correctness (m)	Accuracy	Reduction (%)	Uncertainty Region (m^2)	Certainty	Spatial Certainty	Degree of Anonymity (%)
200	84.36	29.10	15	87.40	318.03	1.91	0.59	29.06
400	75.23	60.49	20	94.12	448.42	2.18	1.02	27.82
600	66.65	96.64	25	96.27	573.95	2.38	1.20	27.69
800	60.30	137.31	28	97.30	635.30	2.46	1.38	26.81
1000	54.36	180.30	30	97.93	699.12	2.54	1.52	26.69
1200	48.20	221.07	31	98.33	744.40	2.57	1.63	26.19
1400	43.33	266.53	32	98.59	787.83	2.63	1.72	26.08
1600	39.31	319.41	33	98.81	788.23	2.62	1.77	25.39

Table 6: Countermeasure privacy metrics.

Defence	Success Rate (%)	Correctness (m)	Accuracy	Reduction (%)	Uncertainty Region (m^2)	Certainty	Spatial Certainty	Degree of Anonymity (%)
Baseline (no defense)	77.26	27.11	14	96.67	267.75	1.78	0.24	22.79
C1 - Generalization: 50 m	76.36	29.46	13	96.64	256.72	1.59	0.34	20.26
C1 - Generalization: 100 m	65.15	38.89	11	96.67	222.42	1.43	0.39	18.03
C1 - Generalization: 200 m	43.24	56.92	10	96.72	198.94	1.28	0.37	16.22
C1 - Generalization: 300 m	27.58	84.80	9	96.91	184.09	1.21	0.39	15.56
C1 - Generalization: 400 m	15.92	118.28	9	96.97	182.07	1.21	0.40	15.33
C1 - Generalization: 500 m	8.58	164.74	8	97.20	170.75	1.16	0.35	14.83
C1 - Generalization: 1000 m	3.04	290.38	6	97.95	133.72	1.03	0.42	12.73
C2 - Noisy distances: 50 m	78.09	31.66	22	96.05	411.56	2.26	0.39	29.07
C2 - Noisy distances: 100 m	78.05	40.77	32	94.90	638.75	2.65	0.69	34.23
C3 - Shifting distances: 50 m	78.13	31.21	22	96.02	417.88	2.28	0.40	29.26
C3 - Shifting distances: 100 m	77.75	41.15	33	94.90	637.14	2.66	0.69	34.12