Watch Nearby!
Privacy Analysis of the *People Nearby* Service of Telegram

Maurantonio Caprolu
maurantonio.caprolu@kaust.edu.sa
RC3 Center, CEMSE Division
King Abdullah University of Science and Technology (KAUST)
Thuwal, Kingdom of Saudi Arabia

Savio Sciancalepore
s.sciancalepore@tue.nl
Eindhoven University of Technology
Department of Mathematics and Computer Science
Eindhoven, Netherlands

Aleksandar Grigorov
a.grigorov@student.tue.nl
Eindhoven University of Technology
Department of Mathematics and Computer Science
Eindhoven, Netherlands

Velyan Kolev
v.kolev@student.tue.nl
Eindhoven University of Technology
Department of Mathematics and Computer Science
Eindhoven, Netherlands

Gabriele Oligeri
goligeri@hbku.edu.qa
Division of Information and Computing Technology, College of Science and Engineering
Hamad Bin Khalifa University
Doha, Qatar

ABSTRACT

People Nearby is a service offered by Telegram that allows a user to discover other Telegram users, based only on geographical proximity. Nearby users are reported with a *rough* estimate of their distance from the position of the reference user, allowing Telegram to claim location privacy. In this paper, we systematically analyze the location privacy provided by Telegram to users of the *People Nearby* service. Through an extensive measurement campaign run by spoofing the user's location all over the world, we reverse-engineer the algorithm adopted by *People Nearby* to compute distances between users. Although the service protects against precise user localization, we demonstrate that location privacy is always lower than the one declared by Telegram (500 meters). Specifically, we discover that location privacy is a function of the geographical position of the user. Indeed, the radius of the location privacy area (localization error) spans between 400 meters (close to the equator) and 128 meters (close to the poles), with a difference of up to 75% (worst case) compared to what Telegram declares. After our responsible disclosure, Telegram updated the FAQ associated with the service. Finally, we provide some solutions and countermeasures that Telegram can implement to improve location privacy. In general, the reported findings highlight the significant privacy risks associated with the use of the *People Nearby* service.

CCS CONCEPTS

• **Security and privacy → Software and application security**; **Web application security**; *Software reverse engineering*.

KEYWORDS

Instant Messaging Apps, Telegram, Localization, Location Privacy

WiSec '24, May 27–30, 2024, Seoul, Republic of Korea

ACM Reference Format:
Maurantonio Caprolu, Savio Sciancalepore, Aleksandar Grigorov, Velyan Kolev, and Gabriele Oligeri. 2024. Watch Nearby! Privacy Analysis of the *People Nearby* Service of Telegram. In *Proceedings of the 17th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '24), May 27–30, 2024, Seoul, Republic of Korea.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3643833.3656121

1 INTRODUCTION

Instant messaging has become an integral part of modern communication, both for personal and professional use [1],[2]. In this context, Telegram is one of the most popular instant messaging applications. It was founded in 2013 and is based on a centralized Cloud-based architecture, with Cloud servers deployed worldwide and the operational center located in Dubai, UAE [3]. According to recent statistics, Telegram accounts for 550 million active users monthly and the average Telegram user spends approximately 3 hours per day using the related mobile cellular app [4].

Security and privacy have been among the selling points of Telegram since its first release [5]. In particular, Telegram supports the privacy of users through several means, including, e.g., encrypted end-to-end chats, device-specific communications, self-destructive messages [6], and in recent years, it has even supported dedicated cryptography contests [7]. Partly due to the perceived enhanced users' privacy, Telegram has been used in various controversial situations, e.g., terrorism and far-right groups, crypto investors, and for the exchange of illegal materials [8],[9].

As a distinctive feature, Telegram offers various social-like services, such as channels and groups. One of such Location-Based Services (LBSs) is *People Nearby*, released to the public in June 2019 [10] [11]. *People Nearby*can be activated by any Telegram user, and it allows users to discover other Telegram users without even being in their contact list and without knowing the telephone number, but only based on geographical proximity. For each user who opts in to participate in the service, Telegram reports the username, a profile photo, and a rough indication of the distance of the remote user from the current user location (see Fig. 1 in Sect. 3). Users can

also create location-based groups, where they can invite anyone in their proximity and exchange messages in groups.

In recent years, the *People Nearby* service of Telegram has been reported in various news related to privacy. Some hobbyists [12] and the news media [13] discussed the threat of triangulating the location of users based on the information displayed in the app, which could affect the privacy of the user. To mitigate such threats, Telegram has updated the functionality multiple times and also released statements delegating privacy issues to the awareness of users who explicitly accept the terms of the app [14]. However, to the best of our knowledge, no scientific contribution systematically investigated the actual location privacy provided to users of the *People Nearby* service. Moreover, no studies provide an in-depth description of how Telegram implements mutual distance computation.

Contribution. In this paper, we carry out a systematic study on the privacy properties of the *People Nearby* service offered by Telegram. Specifically, we provide the following main contributions.

- We design and implement a location-privacy attack to the *People Nearby* service of Telegram exploiting the information available at any client while being able to target any user world-wide.
- We reverse-engineer the methodology used by the *People Nearby* service to report remote users' distances from a given user location. Through an extensive real-world measurement campaign that included 302 independent measurements, we demonstrate that Telegram trades distance accuracy for location privacy, making precise user localization impossible.
- We show that the actual location privacy of *People Nearby* is always less than 500 meters—this one being declared by the service, and this depends on the location of the target user. In particular, while the radius of the uncertainty area at lower latitudes is about 400 meters, the radius of the uncertainty area for users at higher latitudes decreases to about 128 meters—being 25% of the declared value.
- We discuss potential solutions and countermeasures that Telegram can implement either to improve users' location privacy or to make the information displayed to the user consistent with the provided location privacy.

Paper organization. The paper is organized as follows. Sect. 2 reviews related work, Sect. 3 introduces the service *People Nearby* of Telegram, Sect. 5 describes the methodology used to gather the data, Sect. 6 describes our measurement campaign, Sect. 7 shows how the service *People Nearby* works, Sect. 8 analyzes the actual degree of location privacy offered to users of *People Nearby*, Sect. 9 discusses responsible disclosure and ethical aspects of our research, and finally, Sect. 10 draws the conclusion.

2 RELATED WORK

Many free instant messaging applications are available on the market that allow people to communicate with each other through text, voice, and media. Telegram is an open source application with high security features [15] [16], which makes it perfect for scenarios affected by censorship [17, 18] and, unfortunately, for the distribution of illicit content [19, 20].

Telegram has received a lot of attention from the security research community in recent years, especially due to its declared focus on security and privacy. For instance, Ludant et al. [21] identified privacy leaks in Telegram that, combined with the usage of a 5G sniffer, would allow for traffic analysis and stealthy generation of fake traffic to a target user, based only on the availability of the mobile cellular number; Albrecht et al. [5] studied the usage of symmetric cryptography in the end-to-end encryption protocol adopted by Telegram, i.e., MTProto; similarly, Lee et al. [22] provided a security analysis of the end-to-end encryption protocol adopted by Telegram; Nobari et al. [23] carried out an analysis of the messages and the connections of the accounts; Varizipour et al.[24] analyzed the attitude towards privacy of Iranian users of Telegram; Anglano et al. [25] devised a forensic analysis based on the generation of artifacts and their retention in the device storage; Abu-Salma et al. [6] disclosed several design issues of Telegram that impact security, including an unclear description of security features such as the use of encrypted chats; and Barsocchi et al. [26] showed an example of an architecture for LBSs compliant with the European GDPR, using Telegram as an example to implement such a service. Hartle et al. [11] spoofed the position of a smartphone in a warfare scenario (Ukraine), thus allowing the Russian and Ukrainian military forces to reach them using the *People Nearby* Telegram service.

Location privacy has been an increasingly important topic of research due to the proliferation of LBSs and the ubiquitous nature of mobile devices. The ongoing challenge is to balance the utility of LBSs with the preservation of individual privacy, especially in the face of evolving technologies and use cases. In the context of instant messaging applications and services, Li et al. [27] used trace-based analysis to study how real-world users share privacy-sensitive location information. They found that user privacy concerns are correlated with age, sex, mobility, and geographic regions. Privacy threats associated with geosocial networks are discussed by Vicente et al. [28], investigating aspects such as location, absence, co-location, and identity. Wei et al. [29] introduce a privacy attack in which the adversary uses historical movements and friendship information to estimate the user's trajectory. The authors also proposed a solution that allows a user to upload fake locations to protect his privacy. Furthermore, Huaxin et al. [30] recently introduced a privacy attack that combines different mobile social networks, able to predict demographic attributes of users.

To the best of our knowledge, the only scientific contribution investigating location privacy issues in proximity-based LBSs is the one by Ding et al. [31]. The authors investigated location privacy issues that affect a service similar to *People Nearby* offered by *WeChat* and discovered that users could be localized with meter-level accuracy by using simple trilateration from any location in the world. Due to the location obfuscation implemented by Telegram (specifically, the usage of squared grids and the noisy transition boundaries among grids), trilateration approaches cannot precisely locate users of Telegram *People Nearby* (see Sect. 6). Furthermore, compared to the contribution by Ding et al., we reverse-engineered the *People Nearby* service of Telegram, and we show that the location privacy claims made by Telegram are not consistent with the actual location privacy provided by the user. Finally, we argue that Telegram is much more widely distributed worldwide than WeChat,

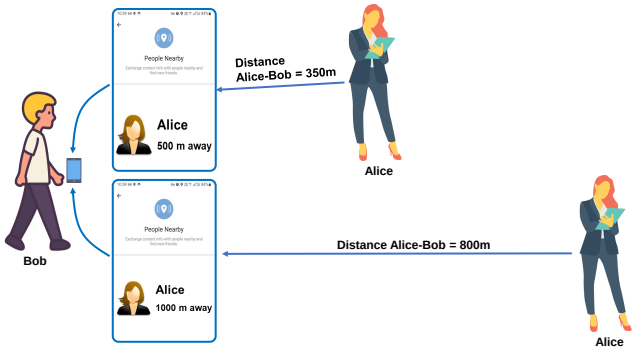


Figure 1: *People Nearby* service provides a rough approximation of the distance between the users in the neighborhood.

contributing to magnifying the potential impact of our research. It should also be noted that although research has already discussed weaknesses and privacy issues related to proximity-based LBSs, the current level of location privacy offered by such services is still very limited, if any.

To the best of our knowledge, no scientific contributions have provided an in-depth analysis of the privacy associated with the service *People Nearby* offered by Telegram.

3 PEOPLE NEARBY SERVICE

People Nearby is a service offered by Telegram that provides the user with a list of other nearby Telegram users while reporting their (rough) distance, as shown in Fig. 1. Furthermore, *People Nearby* also lists location-based groups, i.e., groups associated with a particular geographical location that are visible only to nearby users. Note that location-based groups are different from Telegram *supergroups*, e.g., the ones discussed in [8], which instead are not tied to any location. Telegram provides open-source Application Program Interfaces (APIs) to allow third-party developers to create custom client applications. In this manuscript, we use the Telegram Database Library (TDLib) library[1], which is a cross-platform Telegram client to build custom apps inside the Telegram platform. Specifically, we consider the *searchChatsNearby* API call, which returns a list of users and location-based groups nearby. The function takes the latitude and longitude of the user as input and returns a list containing users and groups located at different classes of distance. The returned list contains up to 100 users, while the list of groups has at most 9 entries. Each entry (being a user or a group) has an associated *distance class* among the following ones: 100 m, 500 m, 1,000 m, 2,000 m, 3,000 m, 4,000 m, 5,000 m, 6,000 m, 7,000 m, 8,000 m, 9,000 m, 10,000 m, 11,000 m, and 12,000 m. The lists are then sorted by distance from the input location. Note that the returned distance class does not allow immediate accurate localization of the nearby user, and this work focuses on analyzing the location privacy provided by the service. The information contained in the lists obtained by using the *searchChatsNearby* function can be used to generate further queries about specific users and groups. For this work, we investigate location privacy taking into account the distance classes of 500 meters and 1,000 meters. For the sake of completeness, we highlight that distances of 100 meters are reported within the service only for users in the user's contact list, whereas higher distances, i.e., higher than or equal to 2,000 meters, are not considered in this work.

Finally, note that our analysis was also restricted by a set of limitations related to the usage of TDLib and the maximum query rate allowed by the *People Nearby* service. Specifically, *People Nearby* sets a maximum daily number of queries per user equal to 1,000, meaning that a user cannot execute more than 1,000 queries per day. When such a limit is exceeded, *People Nearby* gives a temporary *ban* to the user, i.e., it cannot query the system for the next 24 hours. Also, a sudden change in the queried locations among very short consecutive time instants leads to a temporary ban. We empirically set a threshold of 90 km/h for the maximum speed tolerated by *People Nearby*, i.e., if the change in the queried location among consecutive time instants leads to a speed estimation exceeding such a threshold, the *People Nearby* service bans the user. Due to the nature of these (discovered) constraints, we believe that such countermeasures have been enforced by Telegram to mitigate denial-of-service (DoS) attacks and not to protect users' location privacy.

4 ADVERSARY MODEL AND ASSUMPTIONS

In this section, we outline a privacy attack targeting Telegram users enabled by the methodology detailed in Section 5. The main objective of the attacker is to geolocalize, as precisely as possible, a user of the *People Nearby* service. Looking at the interactions with the *People Nearby* service, the adversary has the same capabilities as any other legitimate Telegram user. The attacker interacts exclusively with the *People Nearby* service through standard queries offered in the TDLib library. Consequently, the attacker is subject to any restrictions or limitations imposed by Telegram on any user, e.g., the maximum number of queries allowed over time. In summary, the attack is performed as follows. Initially, the adversary selects a target from the group of users who use the *People Nearby* service. Subsequently, the adversary queries Telegram from various geographical positions to discover coordinates where the distance reported to the target changes. Finally, the attacker estimates the position of the target, as shown in sections 5 and 6. We stress that the attacker is not in the target's contact list. Consequently, even though Telegram reduces the reported distance to 100 meters for mutual contacts, the adversary cannot exploit this knowledge. In the following, we list other general assumptions we considered in our investigation. *People Nearby* is designed for static users, with the aim of connecting people in close proximity for potential chats and friendships. It is unsuitable for users in motion, as the service may malfunction or even result in a ban if kept active while moving, especially at high speed. For small and slow movements (compatible with people moving within a closed environment, such as an office or a house), the behavior of the app is the same as for static users (for example, sitting on a desk), since the granularity of the user's position considered by Telegram is relatively high and does not change for such small distances. For this reason, in our experiments, we consider static targets.

[1]https://core.telegram.org/tdlib

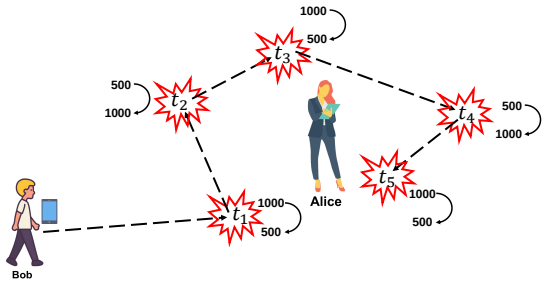


Figure 2: Data acquisition model: $\mathcal{B}$ repeatedly changes his trajectory when he experiences a change in the reported distance class to $\mathcal{A}$, as shown by the *People Nearby* service.

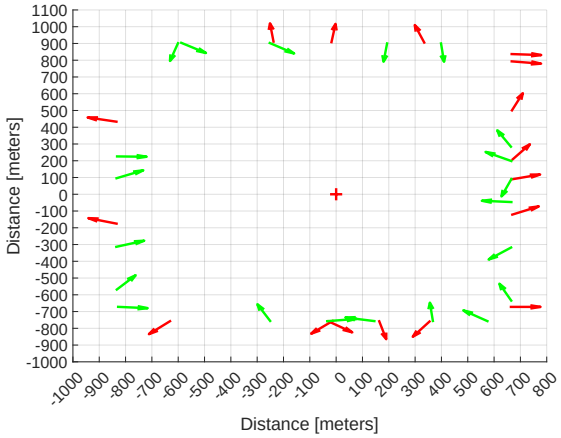


Figure 3: Collection of transitions: red arrows are related to transitions between 500 and 1,000 meters while green arrows report the transitions between 1,000 and 500 meters. The tips and the tails of the arrows refer to the (two) locations—before and after—the transition is detected.

5 METHODOLOGY

In the following, we provide an in-depth analysis of the location privacy provided by the Telegram *People Nearby* service. To carry out such an analysis, we systematically collected data on distance changes (*transitions*) reported by Telegram for specific accounts under our control.

Definition 5.1. We define a *transition* (t_x), with $x \in [0, \ldots, \infty]$, as a change in the distance reported by the Telegram *People Nearby* service from a remote user, as the result of a movement of the local or remote user.

For our analysis, we consider the data acquisition model in Fig. 2, with two users, $\mathcal{A}$ (static) and $\mathcal{B}$ (moving), acting as the *target* and the *finder*, respectively. Specifically, we investigate the transitions between 500 meters and 1,000 meters (and back) generated by $\mathcal{B}$ moving around $\mathcal{A}$. The finder ($\mathcal{B}$) wants to disclose $\mathcal{A}$'s position, while $\mathcal{A}$ wants to keep her position secret or, worst case, with the same uncertainty level claimed by the *People Nearby* service. In our data collection model, $\mathcal{B}$ walks different trajectories, repeatedly moving closer and farther from $\mathcal{A}$. In particular, $\mathcal{B}$ changes his trajectory (and direction) when he experiences a change in the distance between himself and $\mathcal{A}$ as reported by the *People Nearby* service, independently of the transition being 1,000 to 500 or the opposite. Through this strategy, we collect a set of geographical coordinates, i.e., those where we experience a change in the distance between $\mathcal{A}$ (static) and $\mathcal{B}$ (moving).

Figure 3 reports the results of an exemplary real measurement. The red cross at the center $[0, 0]$ of the figure indicates the position of the target: all the geographical coordinates used in the real experiment are converted into the reference system of the target. Recalling the data collection model in Fig. 2, we performed 476 queries, collecting 38 transitions. Note that not all the queries actually collect useful transitions, since we need multiple steps to collect each single transition. The red arrows identify the transitions between 500 and 1,000 meters, while the green arrows denote the transitions between 1,000 and 500 meters. The direction of the arrows is meaningful and shows the actual trajectory of the finder. Finally, we highlight the importance of collecting the transitions in the whole surrounding of the target—this will become clear in the following sections. After discovering a transition, we compute the subsequent trajectory by setting a random direction. However, not all trajectories might go in a useful direction; thus, we empirically defined a threshold to reset the algorithm when the finder moves far away from the target. Without loss of generality, in the following, we only consider the transitions between 500 and 1,000 meters (and back), i.e., both the green and red arrows in Fig. 3. The results of Fig. 3 highlight the existence of a rectangular shape identified by the transitions, while an in-depth analysis of the transition distribution will be provided later. We denote by $\mathcal{T} = [t_o, \ldots, t_N]$ the set of identified transitions. We approximate their positions by considering the rectangular shape identified by the coordinates $[x_m, x_M, y_m, y_M]$, as per Eq. 1.

$$\begin{aligned} x_m &= \min_x(\mathcal{T}) \\ x_M &= \max_x(\mathcal{T}) \\ y_m &= \min_y(\mathcal{T}) \\ y_M &= \max_y(\mathcal{T}) \end{aligned} \tag{1}$$

We show the results of our analysis in Fig. 4. As in the previous case (Fig. 3), we consider all coordinates in the reference system of the target position, i.e., the red cross in the center ($[0, 0]$), while we report the edges of the rectangular shape, identified according to Eq. 1, with the black circles and the centroid associated with the computed edges with the red circle. Note that the centroid is the best approximation of the target position so far—assuming the position of the target is unknown to the finder.

5.1 Data Collection

To implement the data acquisition model depicted in Fig. 2, we used a Samsung S10 smartphone impersonating $\mathcal{A}$, and a DELL

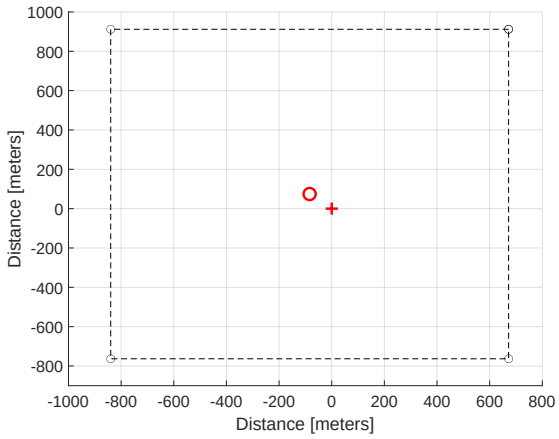


Figure 4: Modelling the transitions' distribution with a square shape: the red cross represents the position of the target, the black circles indicate the corners of the transitions' boundaries, and finally, the red circle shows the position of the centroid, being the best estimation of the target position assuming the target location is unknown to the finder.

XPS 15 laptop impersonating $\mathcal{B}$. The smartphone runs Android 12 and Telegram v9.6.6 (3362), while the laptop runs Ubuntu 22.04 and a home-made script to query Telegram *People Nearby* through the TDLib APIs. First, we place $\mathcal{A}$ in a particular geographical position by spoofing the GPS of the smartphone using the app *Fake GPS Location Professional*[2]. In this way, the *People Nearby* service uses such a fake location as the one of $\mathcal{A}$. Then, we designed and implemented an algorithm to emulate the movement of $\mathcal{B}$ around $\mathcal{A}$, with the aim of collecting as many *transitions* points as possible. The algorithm iterates the following steps: (i) choosing a position for $\mathcal{B}$, (ii) querying Telegram *People Nearby* from that position using the TDLib API (specifically, the function *searchChatsNearby*), and (iii) retrieving $\mathcal{A}$'s distance from the returned list of $\mathcal{B}$'s nearby users. According to the position of $\mathcal{B}$, the algorithm starts at a random position close to $\mathcal{A}$, where the distance from $\mathcal{A}$ is 500 m—consequently, $\mathcal{B}$ is within $\mathcal{A}$'s transitions shape. Then, to choose the next position, it applies the following steps:

(1) **Choose the direction.** Starting from its current position, the algorithm finds a direction where it can move inside the shape while approaching its boundary. This is done by picking a random direction and checking if, after the jump, the new position remains inside the shape, i.e., the distance reported from $\mathcal{A}$ in *People Nearby* remains 500 m. The size of the jump is carefully selected not to violate the limitations in Sect. 3, to avoid our user being banned. When a suitable direction is found, the algorithm moves on to the next phase.
(2) **Probing around the shape.** The algorithm keeps jumping away from the starting point in the direction found in the previous phase. The current phase stops once the distance reported by $\mathcal{A}$ changes from 500 meters to 1,000 meters, i.e., $\mathcal{B}$ falls outside of the shape. Then, the next phase starts from the last identified point (the one outside the shape) and the direction that leads to it.
(3) **Finding the boundary.** In this phase, the algorithm estimates the position of the shape border (boundary) with a predetermined accuracy (in meters). Starting from the transition point identified in the previous phase, the algorithm jumps back and forth over the boundary, halving the jump distance every time to approximate the *transition boundary* with the desired accuracy. In all our experiments, we set such accuracy to 10 meters, to trade-off between the accuracy and the number of queries necessary to find it.

[2]https://play.google.com/store/apps/details?id=com.just4funtools.fakegpslocationprofessional&hl=en_US

6 MEASUREMENT CAMPAIGN

We performed a total of 302 measurements, placing a target in random positions around the world, involving the collection of a total of 8,955 transitions. In the following, we collectively analyze all such measurements to draw conclusions about the distribution of the transition boundaries around the target location, to finally obtain the target localization error in terms of amplitude and phase.

We applied the same analysis as done for Fig. 4, obtaining Fig. 5. All target positions (302) are placed at $[0, 0]$, and as in the previous analysis, we report the edges that approximate the position of the transitions with black circles. At the same time, we compute the red circles as the centroids (mean) of the corners of the squares. Figure 5 shows how the service *People Nearby* works to report the distance of the current user. For each target position, Telegram displaces the transitions in a pseudo-random fashion, although they are aligned in a square shape. The shape of the transitions' boundaries depends on the latitude of the target (see Sec. 8). In the following, without loss of generality, we consider only the case of square shapes. The pseudo-random offset between the square shape and the target prevents precise (meter-level) localization even when collecting many transitions in the surroundings of the target. The location privacy of the target is also shown by the distribution of the centroids (red circles), which are evenly distributed in the target's surroundings. Our subsequent analysis focuses on the distribution of the offset between the target and the transition boundaries.

We now overlap all the centroids at $[0, 0]$, as depicted in Fig. 6. Our analysis shows that the *People Nearby* service maps the location of the users at a pseudo-random position identified by the clouds of the red crosses—we claim the position is pseudo-random since different queries for the same account at different times at the same location return the same transitions. Moreover, note that the transition boundaries (squares) do not have the same size. In particular, we observe a larger deviation on the x-axis (longitude) than on the y-axis (latitude). A manual inspection of the measurements shows that the error might be due to the combination of multiple factors: (i) the random trajectory of our script might not have captured the most "external" transitions as the one depicted in Fig. 3 and Fig. 4, (ii) there was an out of sync between the moving spoofed position and the response to the telegram API request, or simply, an incomplete measurement is taken. Given the previous considerations, we identified a set of measurements that are immune to all identified

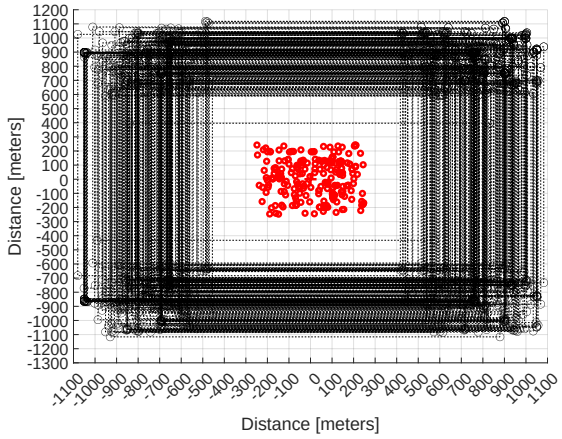


Figure 5: Measurements analysis: the black circles identify the corners of the boundaries of the transition, i.e., the transitions between the distances of 500 and 1,000, meters (in both the ways), red circles show the centroids, while all the targets are overlapping at $[0, 0]$.

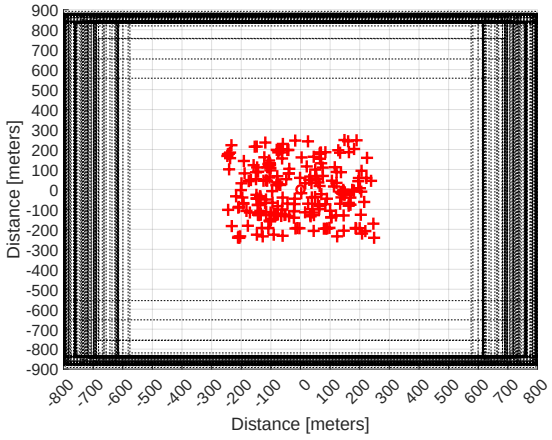


Figure 6: Measurements analysis: We overlap the centroids at the position $[0, 0]$ and the associated squares (transitions between $1,000$ and 500 meters), while the red crosses show the actual position of the targets.

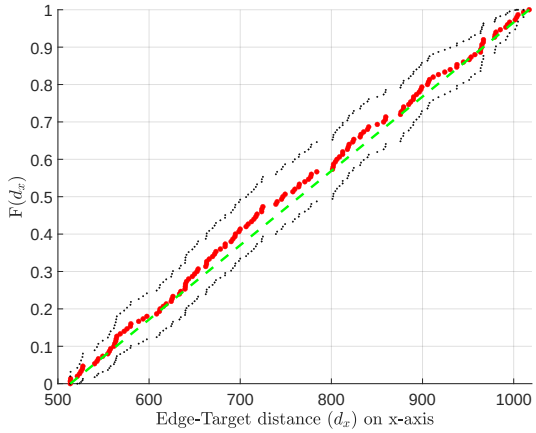


Figure 7: Target position analysis. The cumulative distribution function $F(d_x)$ is calculated on the x component of the targets. The dashed green line shows the associated uniform distribution.

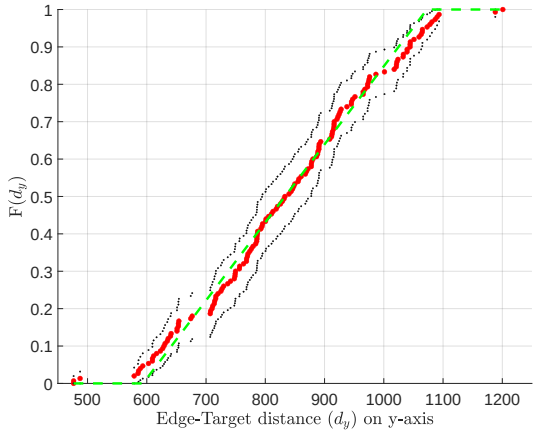
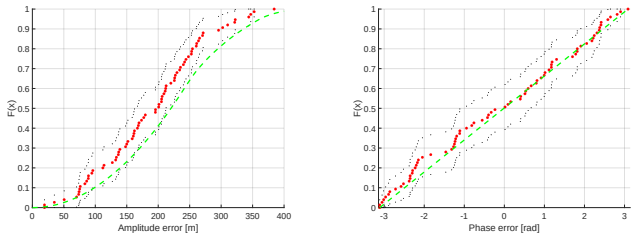


Figure 8: Target position analysis. The cumulative distribution function $F(d_y)$ is calculated on the component y of the targets. The dashed green line shows the associated uniform distribution.

issues, and we continued our analysis. Indeed, we focus on the distribution of a subset of the targets' position in Fig. 6 with respect to the edges (squares). For each pair of coordinates of the target (x, y), we consider the distance between the edges and the coordinates of the target, in terms of x and y, respectively. Figure 7 shows the empirical cumulative distribution function $F(d_x)$ associated with d_x, i.e., the distance between the right edge of the square and the component x of the target coordinate, while black dots show confidence bounds at 0.05. d_x spans between about 513 meters ($F(d_x) = 0$) and about 1003 meters ($F(d_x) = 1$). We obtained similar values, i.e., about 521 and 1010 meters, considering the left edge of the square. Finally, the dashed green line shows the best fit considering a uniform distribution $\mathcal{U}_{[a,b]}$, where a and b are the minimum and maximum of the x coordinates, respectively. We performed the same analysis on the y coordinate, as depicted in Fig. 8. Although there are some outliers, our analysis reveals that the distribution of the y coordinates is also uniform, with bounds of about $[476, 1200]$. Our analysis shows that, given a randomly deployed target, a finder can successfully collect a set of transitions, which in turn provide



(a) (b)

Figure 9: Target localization error in terms of amplitude (a) and phase (b): the relative target position (respect to the transitions' boundaries) is uniformly distributed.

an upper bound on the likelihood of its position. We can compute the uncertainty limits of the location according to Fig. 7 and Fig. 8, being equal to a square of approximately $\approx 513 \times 476$ meters. In addition, we experimentally proved that the position of the target is uniformly distributed inside that area, given the knowledge of the transitions. The approximated position of the target is investigated in more detail in the remainder of this paper, and we will show that it can be much smaller than the one shown by Telegram, i.e., 500 meters.

Finally, we consider the analysis of the amplitude and phase associated with the estimation error of the target location. Recalling Fig. 6, we want to estimate the distance between the centroids (best target position estimation) and the actual target locations. To this aim, we consider the set of phasors identified by the pairs centroid-target and, for each of them, we compute the amplitude and phase. Figure 9 shows the results of our analysis. We confirm that the phase (Fig. 9(b)) is uniform in the range $[-\pi, \pi]$, as also obtained indirectly through our previous analysis. The amplitude can be estimated according to Eq. 2, where ρ is the phasor amplitude, while x and y are uniformly distributed random variables (as per our previous findings).

$$\rho = \sqrt{x^2 + y^2} \tag{2}$$

Red dots in Fig. 9(a) show the empirical cumulative distribution function associated with the amplitude of the phasor. We also reported the confidence intervals (0.05) using the black dots consistent with our previous analysis. The dashed green line depicts Eq. 2 being a good fit for our empirical results. We observe that the location privacy of the target user can be arbitrarily reduced at the cost of precision. As an example, the position of the user can be estimated in a range of about 200 meters with a probability of 0.5. We stress that this analysis is cumulative with respect to all measurements. In the following, we will show that the target estimation error is a function of the target latitude. Therefore, if the adversary is aware of a rough estimation of the target position—as it is reasonable to assume—he can do much better to locate him.

7 REVERSE ENGINEERING *PEOPLE NEARBY*

In this section, we reverse engineer how *People Nearby* works and provide an in-depth analysis of how it achieves location privacy.

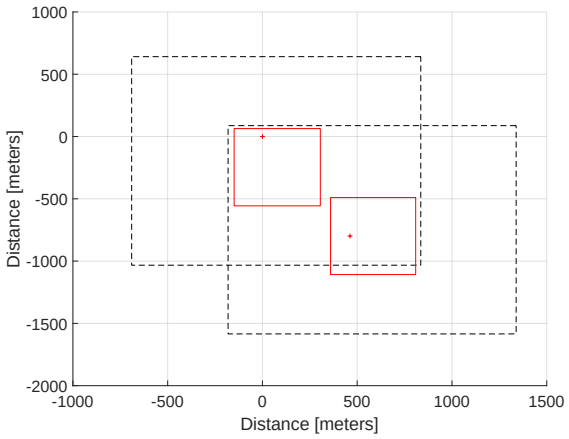


Figure 10: Transition boundaries (dashed black lines) and uncertainty regions (solid red lines) associated with 2 targets (measurements).

We first define the notion of *uncertainty region*, i.e., the regions where users' exact location is hidden, and we show that neighboring uncertainty regions perfectly follow each other while not overlapping.

First, we recall Fig. 6 and consider the boundaries identified by the distribution of the targets (red crosses). In fact, Fig. 6 shows that neighbor targets (red crosses) share the same geographical distribution of the transitions (boundaries identified by the dashed black lines). As discussed in Sect. 6, any target belonging to the cloud represented by the red crosses cannot be uniquely identified, since Telegram generates the same transition boundaries (dashed black lines). Therefore, we compute the boundaries of the *uncertainty region* from all the collected measurements. Figure 10 shows the boundaries associated with both the transitions and the uncertainty regions considering two measurements. All coordinates have been normalized with respect to the position of the target of one of the two measurements. We stress that the black-dashed lines represent the actual boundaries of 2 measurements, while the red lines are the boundaries computed from all the measurements and expressed as their relative distance to the transition boundaries. The toy example of Fig. 10 highlights how the two targets belong to different uncertainty regions, and these regions are adjacent, i.e., non-overlapping on the map. We follow up our analysis by systematically deploying the target on neighbor positions, thus obtaining Fig. 11. We considered a total of 16 measurements (target deployments depicted in Fig. 11 through red crosses). As considered before, for each measurement (target deployment), we report the transition boundaries (black dashed lines) and the uncertainty regions (solid red lines). We observe that the partitioning of the map has a strong symmetry for the transition boundaries and uncertainty regions. In fact, we believe that the (minor) vertical overlap of the uncertainty regions is due to the errors in the measurements and the limited number of collected measurements. We believe that location privacy is implemented through a tessellation of the playground: the map is

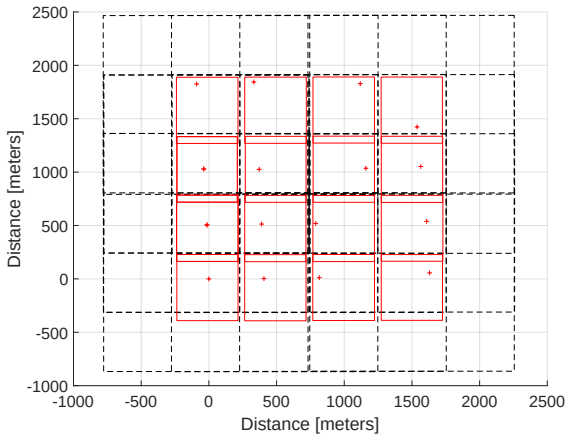


Figure 11: Transition boundaries (dashed black lines) and uncertainty regions (solid red lines) associated with 16 targets (measurements).

divided into adjacent non-overlapping uncertainty regions, and the service returns the associated transition boundaries. Finally, our analysis shows that the transition boundary region has a size of about $1,500 \times 1,600$ meters, while the uncertainty region is about 500×500 meters. It is worth noting that the actual uncertainty region is about one-ninth ($\frac{1}{9}$) of the region identified by the transition boundaries. We stress that these findings are consistent with our previous analysis in Fig. 5 and Fig. 6.

8 LOCATION PRIVACY ANALYSIS

Our final analysis focuses on the actual location privacy provided by the *People Nearby* service. In the following, we compare the upper bound provided by Telegram (recall Sect. 3) with the one that a malicious user can compute by leveraging the Telegram APIs. Indeed, our previous analysis (recall Sect. 7) shows that it is possible to estimate the uncertainty region associated with the user's position; thus, in the following, we compute the upper bound of the localization error. Assuming an uncertainty region of size l meters and the target user in the center of such region, the maximum localization error (upper bound) can be calculated as $\mathcal{D} = \frac{l}{2}\sqrt{2}$ meters. As an example, we recall Fig. 10 and 11. Approximating the square size with $l \approx 500$ meters, we have $\mathcal{D} \approx 354$ meters.

In the following analysis, we consider a set of (target) locations at different latitudes, as shown in Table 1. It is worth mentioning that the calculation of $\mathcal{D}$ for each target location requires multiple measurements. Figure 12 shows our methodology for the estimation of the upper bound $\mathcal{D}$ as a function of the uncertainty region size l. We deployed the same target at different adjacent positions, located 100 meters from each other (as depicted by the red crosses in Fig. 12). Depending on the size of l, multiple measurements may be required to experience a change of the transition boundaries (dashed line) and, therefore, of the uncertainty region (solid red line). When the transition boundary shifts, we consider the size of the shift as the size of the uncertainty region l, thus $\mathcal{D}$ can be

Table 1: Locations considered for the estimation of the upper bound $\mathcal{D}$ associated with the localization error.

City	Country	Latitude	Longitude
Kourou	French Guiana	5.154237	-52.648526
Coban	Guatemala	15.46463	-90.403683
Doha	Qatar	25.26174	51.359269
Lakatamya	Cyprus	35.11438	33.296804
Carcassonne	France	43.21324	2.344961
Winnipeg	Canada	50.00542	-97.16734
Malmo	Sweden	55.555275	13.015577
Helsinki	Finland	60.239339	24.922424
Bodo	Norway	67.277398	14.374172
Utqiagvik	Alaska	71.300602	-156.754113

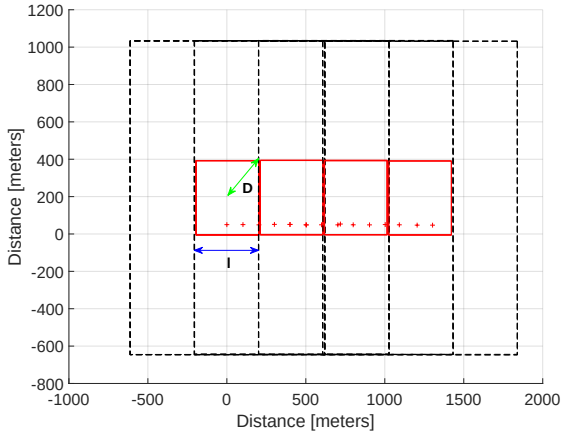


Figure 12: Methodology to compute the upper bound $\mathcal{D}$ associated with the localization error. We estimated the size of the uncertainty region l, and subsequently, we computed $\mathcal{D}$.

computed accordingly. In the (real) example of Fig. 12, l turns out to be about 400 meters, while $\mathcal{D} \approx 282$ meters. Finally, we stress that experiencing 2 shifts may not be enough because of the granularity of the movements of the target, and we also verified the consistency of our estimate of $\mathcal{D}$ over multiple shifts, e.g., 4 shifts in Fig. 12.

We applied the same methodology to different cities at different latitudes around the world, as reported in Table 1. Figure 13 shows the maximum localization error $\mathcal{D}$ as a function of the latitude considering the cities in Table 1. Note that the upper bound on the localization error $\mathcal{D}$ is significantly affected by the target latitude; indeed, it spans between approximately 400 meters at low latitudes (close to the Equator) and about 128 meters at higher altitudes (close to the North Pole). It is worth noting that such distances are about 25% and 75% smaller than what is declared by Telegram, i.e., 500 meters independently of the position of the target, shown by the red dashed line in Fig. 13 (recall Sect. 3). We observe that such a value (500 meters) is not experienced in any of the considered locations, while being 100 meters more than the largest distance

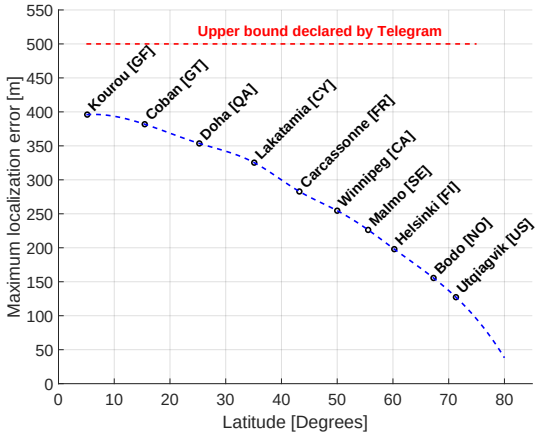


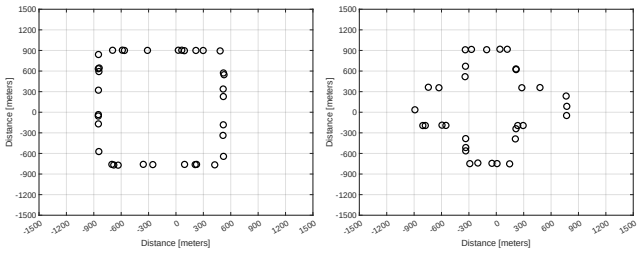
Figure 13: Maximum localization error of the target as a function of its latitude.

experienced, i.e., Kourou (French Guiana), with $\mathcal{D} \approx 400$ meters. We believe that the upper bound associated with the localization error depends on the latitude due to the tiling process performed by the Telegram algorithm (as described in Fig. 11). Our intuition is that the edges of the tiles are linked to the meridians, thus implying tiles with smaller areas when the latitude increases, i.e., closer to the poles. Although our analysis did not consider locations in the southern hemisphere, note that the position of the target could become arbitrarily small when moving north, thus significantly affecting user location privacy, since the uncertainty region is indeed much smaller than the one reported to the user by Telegram.

8.1 Shapes of transition boundaries

In this section, we provide an in-depth analysis of the shapes associated with the transition boundaries as a function of the target location. Figure 14 shows two examples of transition boundaries collected during our measurement campaign. The square shape, i.e., Fig. 14(a), has been previously discussed. Furthermore, as a function of the target position, we encountered a different shape, i.e., a cross like the one in Fig. 14(b). We observe that the two shapes have similar dimensions, i.e., $1,500 \times 1,800$ meters, while for both cases the analysis of the uncertainty region is consistent with the findings reported in Sect. 7. According to our experimental data, spanning over three months of measurements, the shape of the transition boundaries for a particular area is fixed over time.

In the subsequent analysis, we considered all measurements collected around the world and manually analyzed them to judge if the shape reassembles either a square or a cross. Figure 14 shows the results of our analysis, where we draw a blue square or a red cross at the same position of the target as a function of the shape associated with the transition boundary. Our findings show that the shape of the transition boundary is a function of latitude. Indeed, the world map appears split into stripes, where crosses interleave with squares. On the one hand, we stress that the shape does not affect our previous analysis (Sect. 8). Indeed, the size of the uncertainty



(a) (b)

Figure 14: Shapes of the transition boundaries: Our measurement campaign exposes two different shapes for the uncertainty regions, i.e., a square (a) and a cross shape (b).

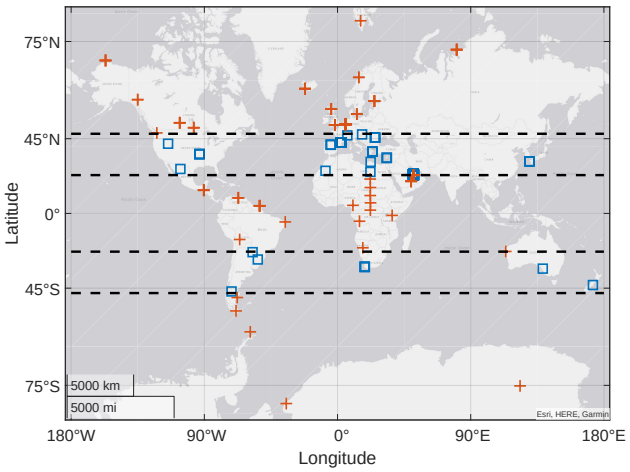


Figure 15: Shape of the transitions boundary: Square and cross patterns as a function of the target's location.

region is the same, independently of the transition boundary shape being a square or a cross. On the other hand, we acknowledge that we cannot provide a full justification for the phenomenon.

9 IMPACT AND RESPONSIBLE DISCLOSURE

Impact of our findings. Our analysis shows that users who activate *People Nearby* experience a location privacy much smaller than the one declared by Telegram, i.e., from 25% (best case) to 75% (worst case) smaller according to the user's latitude.

We recall from Section 2 that Telegram is used a lot worldwide for illegal or controversial activities. On the one hand, legal authorities may exploit the mentioned lack of privacy to identify the actual areas where such activities performed via Telegram are carried out, improving their chance to associate Telegram accounts with people (e.g., using the rough location and profile photo of the user). On the other hand, criminals can also exploit such vulnerabilities, threatening people not only online, but in real life. Therefore, *People Nearby* can cause privacy significantly impacting the lives of users.

Solutions and countermeasures. We propose solutions and countermeasures to make users fully aware of the location privacy provided by *People Nearby*, meet the privacy level declared by Telegram, or make the presented attack harder (or even impossible).

Update the location privacy. A solution involves displaying the actual location uncertainty as the distance reported for each account in the *People Nearby* service, rather than 500 meters (currently displayed). In this way, users can immediately identify what is the current degree of location privacy, and possibly realize that such location uncertainty changes with the current latitude value, as found in our analysis. Another solution consists of displaying, for each user nearby, the smallest location privacy value worldwide, i.e., approximately 128 meters, rather than the 500 meters currently displayed. This is a worst-case approach, declaring to the user the lower bound of location privacy that can be guaranteed worldwide.

Increase location privacy. This countermeasure would require a significant modification of the *People Nearby* service and would consist of adopting tiles with areas as a function of the current latitude, i.e., larger tiles at higher latitude values, to effectively provide an uncertainty region of 500 meters to any user, regardless of their current location. Such a modification could require a large system update of the *People Nearby* service, but would contribute to guaranteeing the extent of location privacy promised to users.

Change API policies. The attack described in this paper relies on one client, i.e., the finder, moving around another user, i.e., the target, in a relatively short time frame. To mitigate this threat, we propose restricting service usage to the first geographic location declared by the user in a given time. The API could be configured to answer users' requests within a limited area, e.g., 100 square meters while dropping requests from more distant locations. This modification allows legitimate users to move within a limited area while using the service. The allowed area can be updated after a relatively short time, for example, every 10 minutes, to allow users to move consistently with the intended use of the service. On the contrary, quickly transitioning between positions 500 meters apart, as our algorithm does, would become unfeasible. Implementing this countermeasure is simpler than the previous one, because it primarily involves request filtering. The service's fundamental logic remains intact, while the attack vector is effectively neutralized. In addition, this countermeasure will not affect legitimate clients, as the service is not intended for users on the move.

Responsible disclosure. Transparency and ethics are of paramount importance to the authors. As such, before disclosing to the public our findings, we practiced the principle of responsible disclosure to ensure the safety and integrity of the affected parties. When identifying the lack of privacy associated with *People Nearby* as discussed in this research, we reached out to Telegram. This was done to provide them with a comprehensive understanding of the issue and the time to take the necessary corrective measures or implement preventive strategies. The purpose was to minimize the possible harm or misuse of the information presented in our study.

After a few weeks, Telegram acknowledged our methodology by highlighting that *"the coordinates of all points are always rounded... It is approximately 556 meters in length and width (or 787 meters diagonally) when close to the equator, which is equal to the 400-meter radius result from your research"*. Moreover, they also pointed out that *"we'll shortly update the FAQ with the relevant information that latitude has a minor effect on these distances."* Note that the decision to make our findings public was driven by the commitment to advance knowledge in the field and to ensure that the general community is informed. We firmly believe that transparency, when combined with responsibility, can drive positive change, fostering a safer and more secure digital environment.

10 CONCLUSION

We have conducted a systematic analysis of the privacy of users of the *People Nearby* service featured by Telegram. We reverse-engineered the algorithm used by *People Nearby* to display rough distances between users, and experimentally showed that the actual location privacy is always less than the reported one of 500 meters. Moreover, location privacy also decreases while increasing the geographical latitude of users. In particular, while the radius of the uncertainty area declared by Telegram is 500 meters, our analysis shows that such a radius spans between 400 meters and 128 meters when the user location is close to the equator and the north pole, respectively. It should be noted that the actual uncertainty region is characterized by a radius that is approximately 25% and 75% smaller than that declared by Telegram as a function of the user's location. We believe that the lack of location privacy generates significant risks for the involved users. Such concerns might motivate users to opt out of using LBSs while calling for further efforts by Telegram to protect users' privacy.

ACKNOWLEDGEMENTS

This research was made possible by the INTERSECT project, Grant ID NWA.1162.18.301, funded by Netherlands Organisation for Scientific Research (NWO). The contents herein are solely the responsibility of the author(s).

REFERENCES

[1] P. Rösler, C. Mainka, and J. Schwenk, "More is Less: On the End-to-End Security of Group Chats in Signal, WhatsApp, and Threema," 2018, pp. 415–429.
[2] R. Alkhulaiwi, A. Sabur, K. Aldughayem, and O. Almanna, "Survey of secure anonymous peer to peer Instant Messaging protocols," in *2016 14th Annual Conf. on Privacy, Security and Trust (PST)*. IEEE, 2016, pp. 294–300.
[3] A. Dargahi Nobari, N. Reshadatmand, and M. Neshati, "Analysis of Telegram, an instant messaging service," in *Proc. of the 2017 ACM on Conf. on Information and Knowledge Management*, 2017, pp. 2035–2038.
[4] "How Many People Use Telegram in 2023? 55 Telegram Stats," https://backlinko.com/telegram-users, 2023, accessed: 07-Feb-2024.
[5] M. Albrecht, L. Mareková, K. Paterson, et al., "Four Attacks and a Proof for Telegram," in *2022 IEEE Symp. on Security and Privacy*, 2022, pp. 87–106.
[6] R. Abu-Salma, K. Krol, S. Parkin, et al., "The Security Blanket of the Chat World: An Analytic Evaluation and a User Study of Telegram," in *Proc. of EuroUSEC*, 2017.
[7] "$300,000 for Cracking Telegram Encryption," https://web.archive.org/web/20190518221133/https://telegram.org/blog/cryptocontest, 2019, accessed: 07-Feb-2024.
[8] M. L. Morgia, A. Mei, A. M. Mongardini, and J. Wu, "It's a Trap! Detection and Analysis of Fake Channels on Telegram," in *2023 IEEE Int. Conf. on Web Services (ICWS)*, 2023, pp. 97–104.
[9] "Telegram Receives Massive Fine For Failing To Report Illegal Content," https://www.digitalinformationworld.com/2022/10/telegram-receives-massive-fine-for.html, 2022, accessed: 07-Feb-2024.
[10] "Telegram adds location-flavored extras and full group ownership transfers," https://tinyurl.com/3hv42cn4, 2019, accessed: 07-Feb-2024.
[11] F. X. Hartle, M. Garfinkel, D. O'Neil, G. Scalise, N. Sauer, and C. Willard, "The impact of social media geolocation on national security and law enforcement." *Issues in Information Systems*, vol. 23, no. 1, 2022.
[12] "Telegram Nearby Map," https://github.com/tejado/telegram-nearby-map, 2023, accessed: 07-Feb-2024.

[13] "Don't Use Telegram's New 'People Nearby' Feature," https://lifehacker.com/dont-use-telegrams-new-people-nearby-feature-1846017886, 2023, accessed: 07-Feb-2024.
[14] "Telegram's People Nearby feature reveals exact user locations through triangulation," https://www.androidpolice.com/2021/01/05/telegrams-people-nearby-feature-reveals-exact-user-locations-through-triangulation/, 2021, accessed: 07-Feb-2024.
[15] T. Sutikno, L. Handayani, D. Stiawan, M. A. Riyadi, and I. M. I. Subroto, "Whatsapp, viber and telegram: which is the best for instant messaging?" 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:61685613
[16] T. Sušánka and J. Kokeš, "Security analysis of the telegram im," ser. ROOTS. New York, NY, USA: Association for Computing Machinery, 2017. [Online]. Available: https://doi.org/10.1145/3150376.3150382
[17] S. Kargar and K. McManamen, "censorship and collateral damage: Analyzing the telegram ban in iran," *Berkman-Klein Center Research Publication*, no. 4, 2018.
[18] K. Ermoshina and F. Musiani, "The telegram ban: How censorship "made in russia" faces a global internet," *First Monday*, vol. 26, no. 5, 2021.
[19] J. Guhl and J. Davey, "A safe space to hate: White supremacist mobilisation on telegram," *Institute for Strategic Dialogue*, vol. 26, 2020.
[20] S. Walther and A. McCoy, "Us extremism on telegram," *Perspectives on Terrorism*, vol. 15, no. 2, pp. 100–124, 2021.
[21] N. Ludant, P. Robyns, and G. Noubir, "From 5G Sniffing to Harvesting Leakages of Privacy-Preserving Messengers," in *2023 IEEE Symp. on Security and Privacy (SP)*, 2023, pp. 3146–3161.
[22] J. Lee, R. Choi, S. Kim, and K. Kim, "Security analysis of end-to-end encryption in telegram," in *Simposio en Criptografía Seguridad Informática, Naha, Japón. Disponible en https://bit. ly/36aX3TK*, 2017.
[23] A. Dargahi Nobari, N. Reshadatmand, and M. Neshati, "Analysis of telegram, an instant messaging service," in *Proc. of the 2017 ACM on Conf. on Information and Knowledge Management*, 2017, p. 2035–2038.
[24] E. Vaziripour, J. Wu, R. Farahbakhsh, K. Seamons, M. O'Neill, and D. Zappala, "A Survey of the Privacy Preferences and Practices of Iranian Users of Telegram," in *Workshop on Usable Security (USEC)*, vol. 1, 2018.
[25] C. Anglano, M. Canonico, and M. Guazzone, "Forensic analysis of telegram messenger on android smartphones," *Digital Investigation*, vol. 23, pp. 31–49, 2017.
[26] P. Barsocchi, A. Calabrò, A. Crivello, et al., "A Privacy-By-Design Architecture for Indoor Localization Systems," in *Int. Conf. on the Quality of Information and Communications Technology*. Springer, 2020, pp. 358–366.
[27] N. Li and G. Chen, "Sharing location in online social networks," *IEEE Network*, vol. 24, no. 5, pp. 20–25, 2010.
[28] C. Ruiz Vicente, D. Freni, C. Bettini, and C. S. Jensen, "Location-related privacy in geo-social networks," *IEEE Internet Computing*, vol. 15, no. 3, pp. 20–27, 2011.
[29] W. Chang, J. Wu, and C. Tan, "Friendship–based location privacy in mobile social networks," *Int. Journal of Security and Networks*, vol. 6, no. 4, pp. 226–236, 2011.
[30] H. Li, H. Zhu, S. Du, X. Liang, and X. Shen, "Privacy leakage of location sharing in mobile social networks: Attacks and defense," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 646–660, 2018.
[31] Y. Ding, S. T. Peddinti, and K. W. Ross, "Stalking Beijing from Timbuktu: A Generic Measurement Approach for Exploiting Location-Based Social Discovery," in *Proc. of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*, ser. SPSM '14, 2014, p. 75–80.