Spring 2022

Achieving Location Privacy in iOS Platform Using Location Privacy Framework

Anna Systaliuk
San Jose State University

ACHIEVING LOCATION PRIVACY IN IOS MOBILE PLATFORM USING LOCATION PRIVACY FRAMEWORK

A Thesis

Presented to

The Faculty of the Department of Computer Engineering

San José State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Anna Systaliuk

May 2022

The Designated Thesis Committee Approves the Thesis Titled

ACHIEVING LOCATION PRIVACY IN IOS MOBILE PLATFORM USING LOCATION PRIVACY FRAMEWORK

by

Anna Systaliuk

APPROVED FOR THE DEPARTMENT OF COMPUTER ENGINEERING

SAN JOSÉ STATE UNIVERSITY

May 2022

Wencen Wu, Ph.D.	Department of Computer Engineering
Kaikai Liu, Ph.D.	Department of Computer Engineering
Nima Karimian, Ph.D.	Department of Computer Engineering

ABSTRACT

ACHIEVING LOCATION PRIVACY IN IOS MOBILE PLATFORM USING LOCATION PRIVACY FRAMEWORK

by Anna Systaliuk

Rising popularity of location-services mobile applications and geotagging digital activities resulted in astonishing amount of mobility data collected from user devices, raising privacy concerns regarding the way this data is extracted and handled. Despite numerous studies concluded that human location trace is highly unique and poses great re-identification risks, modern mobile operating systems fell short of implementing granular location access mechanism. Existing binary location access resulted into location-based-services being able to retrieve precise user's coordinates regardless of how much details their functionality actually require and sell it to data brokers. This paper aims to provide practical solution how a mobile operating system (iOS) can adopt a system that enforces better location privacy for user devices with Location Privacy Framework(LPF) that works as a trusted middleware between mobile operating system and third-party apps. LPF provides granulated way of extracting location-related data from device, maximizing privacy by applying geomasking algorithm based on minimum level of accuracy the app needs and ensuring k-anonymity with dummy-generation mechanisms. Furthermore, LPF enforces control over all location data network communication to and from the app to make sure that no identifying data is being shared with data brokers.

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Gopinath Vinodh, that lead me throughout the process of thesis writing, patiently guiding me in formulating research questions and methodology approach. Your support and belief in me helped tremendously in my work, academic and career pursuits.

I am grateful for Committee Members Dr. Wencen Wu, Dr. Kaikai Liu, and Dr. Nima Karimian for providing your valuable expertise, feedback and insightful comments, and letting my thesis defense be such a great experience.

I would like to thank Eyad Murshid for his professional mentorship and guidance in wonderful app design and implementation. Witnessing your exceptional talent and masterliness inspired me to work harder and push my limits every step of the way.

Finally, I would love to express my gratitude to my family and friends, who were there for me though all the difficult times. Your constant love and support made it possible for me to accomplish everything I have today.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 Location Privacy

Modern technologies and widespread usage of smartphone apps introduced many significant improvements into our lives as we are able to enjoy innovative forms of entertainment, take advantage of new types of services and connect with one another with ease. Mobile devices store up an astounding amount of personal data - such as contacts, pictures, messages, payment methods information, shopping history, passwords, and location data. Much of this information is being shared with mobile apps to provide users with personalized service. However, it also introduced numerous privacy threats, such as user profiling, exposing sensitive information, and re-identification.

In a recent "Day in The Life of Your Data" document [1], Apple gives an example of a privacy threat situation illustrated by a real-life scenario. The narrative starts with a setting where a father spends a day hanging out with his daughter. Throughout which they perform a sequence of innocent activities such as looking up the weather, using a navigation app to pick up his daughter from school, taking a selfie with a filtering app and posting it on social media, and buying a toy on the way home. At the end of the day, sensitive information such as family's home address, location of the park and shops they visited, and purchases history was gathered by third parties by data tracked across apps used to perform each activity without user's awareness. These apps track family activities every day and report new information to tracking companies to update profiles that they keep on each user. With this amount of information available, tracking companies could easily reveal the identity of father and daughter and sell their mobility trace to any third parties who are willing to pay for it, putting the family's privacy and safety at risk.

According to recent research done by Binns, Lyngs, Kleek, Jhao, Libert, and Shadbolt [2], the median number of trackers found in mobile apps is ten. Furthermore, their analysis revealed that 90.4% of mobile apps observed in research shared user

information with at least one of 5 dominant tracking companies. These data brokers harvest data they collect about each user across multiple apps to build a demographic and behavioral profile that can be sold to the highest bidding third parties and be monetized by targeted advertising. Data brokers may not be able to identify users they collect information on at first. With time, they could reveal the identity of users by accumulating enough information across multiple apps in user profiles, putting people's privacy and safety at risk. Just because people are not aware of how their personal data is being sold, it doesn't mean that this does not affect them - as illustrated in 2018 major data breach revealing Cambridge Analytica using Facebook to access people's profiles, locations, friends without their consent and show targeted ads to persuade people's political opinions in 2016 Presidential Election.

In this paper, I will be focusing on location privacy - users' right to their current location and mobility trace to remain anonymous for third parties. In their recent research, Montjoye, Hidalgo, Verleyse, and Blondel calculated that "at most eleven points are enough to uniquely characterize all considered traces that are required to identify an individual" [3]. Since location data contains highly sensitive information - tracing a person's movement across time, revealing users' frequent locations, such as home or office addresses, and provides a high chance of person re-identification. The way location data is currently being handled by mobile operating systems and location applications leaves many loopholes for adversaries to gain access to identifiable information that could reveal personal data.

1.2 Motivation

Before the smartphone era, the majority of user location and mobility trace data was only available to phone carriers and was only limited to cell tower data that a particular mobile activity was connected to. Nowadays, as smartphones are becoming more technologically sophisticated, they are able to record and share a staggering amount of

personal data. Location is most commonly shared by user's mobile device and being collected by location-based service software used in navigation (Google Maps), gaming (Pokemon Go), infotainment (Netflix), searching (Yelp), shopping (Amazon), and social media applications that enable users to share their live location with other users, find places nearby, and engage in a lot of different activities. Notably, because people usually carry their mobile devices with them everywhere they go, the amount of mobility and location data that became available has been rapidly growing. Location-based mobile applications are able to collect a person's geolocation on a daily basis and send it to data brokers that keep track of all the data they can get about each device in user profiles.

This widespread availability of mobile data raised concerns about the limits of human privacy regarding location data. A number of studies that have been conducted to study the nature of human mobility data indicated that human mobility data is highly distinctive and can potentially reveal a lot of sensitive private information about people. Montjoye, Natgunanathan, Verleysen, and Blondel conducted a study that examined mobility data for 1.5 million people for 15 months to find patterns that might be helpful in understanding the nature of data. They concluded that "knowing as few as four Spatio-temporal points were taken at random (Ip = 4) is enough to uniquely characterize 95% of the traces amongst 1.5 M users" [3]. Because of the high uniqueness of human mobility trace, location data should be protected for users to maintain anonymity and identity security.

More users are becoming aware of how much of their personal data is being collected and possible privacy threats, demanding more transparent and secure ways of collecting and handling their data, pushing governments around the world to take action by passing new laws that regulate data collection with laws like General Data Protection Regulation (GDPR) passed by EU in 2018, Location Privacy Protection Act 2014, California Privacy Rights Act 2020, etc. In a user privacy preferences study conducted by Fawaz and Shin [4], 180 smartphone users were surveyed to determine location privacy

preferences. According to their findings, “78% of the participants believe that apps accessing their location can pose privacy threats . . . and 52% of the surveyed individuals stated no problem in supplying apps with imprecise location information to protect their privacy” [4]. It is evident that large number consumers are even willing to compromise on decreased functionality for data privacy protection.

In order to address these concerns and maintain user’s trust, companies that collect, store, and distribute user data should work towards adjusting their business practices and researching new privacy-preserving data collection and handling methods. This paper explores mechanisms and technologies that can be used to improve location data extraction process from user device, introducing ways iOS mobile operating system can implement these type of changes to guarantee more privacy for users using third-party location apps downloaded from App Store platform.

This paper explores existing techniques that used to improve user location privacy in iOS. To showcase how existing mobile solutions could integrate better geoprivacy practices, this paper provides background information of existing privacy measures, geomasking methods, and how location data is handled in iOS devices from user’s and developer’s perspective in Chapter 2. Chapter 3 discusses related works addressing Location Privacy Solution Mechanisms, Privacy Issues in Mobile Platforms, and Location Privacy Solutions in Mobile Platforms. Chapter 4 describes solution design and how Location Privacy Framework interact with other components of the system, anonymization algorithm used and use cases. Chapter 5 gives technical details of implementation of experimental work and its results. Finally, Chapter 6 and Chapter 7 includes paper content summary and discussion of possible solution improvements and geoprivacy techniques integration into existing iOS system.

2 BACKGROUND

2.1 Privacy Metrics

"Shockingly, there remains a common incorrect belief that if the data looks anonymous, it is anonymous."

— L. Sweeney

2.1.1 K-Anonymity

Anonymization goes far beyond removing obvious identifiers, such as full name, social security or ID number. Some characteristics that would not reveal the identity of an individual per se, can be combined with other characteristics that result into unique combinations pertinent only to a specific person. For example, one study analyzed publicly available data of 54,805 voters and concluded that 97% of them could be identified just based on postal code and date of birth [5]. In this case, even though data did not directly reveal identity of voters, it was enough to combine few secondary characteristics and some additional public data to identify most of the voters, leaking sensitive data about people's political affiliations. These types of characteristics are called quasi-identifiers and as with most of activities being recorded by mobile devices, number of quasi-identifiers is growing with every new digitalized functionality.

Most commonly used measure of anonymization and data privacy protection is k-anonymity. The concept was first introduced in "Protecting Privacy when Disclosing Information: k-Anonymity and Its Enforcement through Generalization and Suppression" by P. Samarati and L. Sweeney in 1998, and it refers to a level of data anonymization in which each data point with a specific combination of quasi-identifiers is indistinguishable from at least k-1 other data points [6]. When dealing with location data, location latitude, longitude, and timestamp are regarded as quasi-identifiers, since an individual can be identified by with enough spatio-temporal data gathered.

2.1.2 L-Diversity

However, in some cases, achieving k-anonymity is not enough to keep personal data protected. Even if each combination of quasi-identifiers in a dataset is linked to at least k data points, lack of data diversity may result in possibility of homogeneity attack and sensitive information leaking. For example, medical records with k patients with the same quasi-identifier values (zip code, gender, age, etc.) adhere to k-anonymity criteria. But if all those k patients have the same medical diagnosis, then this sensitive information is exposed for all of them! L-diversity helps prevent this type of attack by enforcing an additional requirement of having at least l(el) distinct values for each sensitive attribute for every quasi-identifier attributes equivalence class in a dataset.

2.1.3 T-Closeness

T-closeness criteria extends idea of l-diversity beyond categorical value types. L-diversity guarantees l distinct sensitive values within each quasi-identifier characteristic equivalent class, however if the values are semantically or numerically close, it may still cause sensitive information leaking in a similarity attack. Since location coordinates is quantitative value, if we have l distinct but very numerically close values representing user locations from the same equivalence class, l-diversity would not be enough to keep user data anonymized. This is particularly relevant for large gathering areas and high-density urban places, where it would be easy to locate user's exact location if there are a lot of other people in small proximity to the user.

2.2 Geomasking Methods

The higher the accuracy of shared location, the higher the chance of privacy threats. Level of minimum location precision that would not compromise the quality of location based service varies greatly and depends on the functionality. Geomasking is the process of replacing original spatial datapoint with a new location to protect user privacy.

However, each geomask inevitably result in some spatial resolution loss, so it is important to consider the trade-off between geoprivacy and required location precision when applying geomask. Different geomasking location obfuscating techniques are used based on location service functionality and accuracy requirements.

2.2.1 Aggregation

Most commonly used geomasking technique is aggregation because it provides the highest level of privacy. In point aggregation, all location points within a specific boundaries (like street, city, zip code, country) are mapped to one point in that area (typically determined by geographic center of the region or a specific point-of-interest location), while spatial aggregation maps each point to the whole region. The solution discussed in this paper will use point aggregation approach for aggregated data.

Point aggregation offers high level of user confidentiality at the cost of spatial resolution loss. It's a great privacy solution for location app functionalities that rely on some underlying characteristics of user's current location instead of specified accuracy. For instance, a weather application needs to know which city user is located at to display accurate information, so replacing original location with point aggregated on city-level would guarantee geoprivacy without service quality loss.

However, aggregation geomask is problematic for location services that need finer location resolution or have accuracy constraints. Because geographic regions are not uniform, aggregation cannot guarantee fixed result accuracy. City-level aggregation point for Mendocino, CA with total area of 7.42 square miles will inevitably be more precise than aggregation of Los Angeles, CA with 503 square miles region.

2.2.2 Random Perturbation

Random Perturbation is one of the most straightforward geomasks that does not involve aggregation. Original point is displaced by a random point within a circular region with fixed radius. Because radius does not change based on original location, random

perturbation can guarantee the accuracy of geomasked point. This method reveals more location details than aggregation, so it can be used for more location functionalities that have minimum accuracy requirements.

Because masked point could be located anywhere within a circular region, it could potentially coincide with the original point or be very close to it. This creates a privacy threat of revealing sensitive user location. If adversary gains access to a large dataset with geomasked points, there is a chance that some of them reveal user's original location, making them targets of re-identification.

2.2.3 Donut Method

In random perturbation, there is a chance of location to be geomasked to user's true location. Donut method addresses the shortcomings of random perturbation by adding additional condition of having minimum radius distance between original location and masked point. Each location point is reallocated in a random direction by distance that is bigger than minimum distance and smaller than accuracy limit specified by system, shaping the geomasking region to look like a donut in Fig. 1 This way, point geomasked with donut method maintains a certain level of anonymity guaranteed by minimum radius distance, while preserving accuracy enforced by upper bound radius. Lower radius bound could be a fixed value specified by the system or a fraction of upper bound radius.

In adaptive implementations, outer donut radius is not a constant value - it's computed based on a number of underlying spatial characteristics of user's original location, such as density. Smaller cloaking area in high-density urban places may yield the same anonymization magnitude as larger area in low-density regions, resulting in inversely proportional relationship between geomasking method radius and region density.

Study by Hamption et al. applied different geomasking methods on dataset of medical records and examined their geoprivacy effectiveness. They compared random perturbation with adaptive Donut Method based on population density with administrative boundary

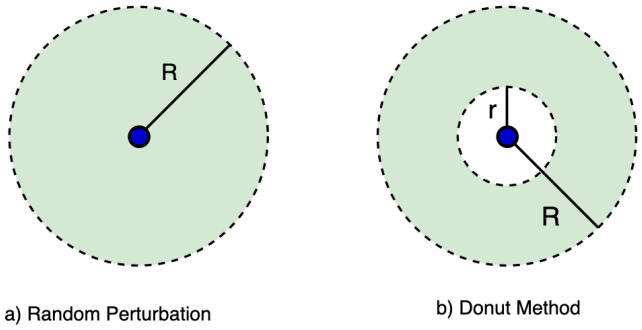

Fig. 1. Anonymization regions for a) Random Perturbation method with radius R; b) Donut Method with lower bound r and upper bound R.

preserving constraint, and concluded that donut method was measured to be 42.7% more effective in protecting privacy and 4.8% lower in cluster detection compared to random perturbation [7].

Privacy Framework solution discussed in this paper uses adaptive donut method in location anonymization algorithm with cloaking region radius defined by minimum accuracy crate of Location Based Service. LPF algorithm did not apply any constraints of preserving administrative or political boundaries (they are not relevant for a large number of location apps) because framework aims to be universal and be able to support as many location services as possible.

2.3 iOS: Location Services

Location Services refers to a framework that integrates location features in iOS devices and provides user control over location data sharing with system services and third-party applications.

iOS user can access Location Services in Settings - Privacy - Location Services. Location Services page is shown in Fig. 2

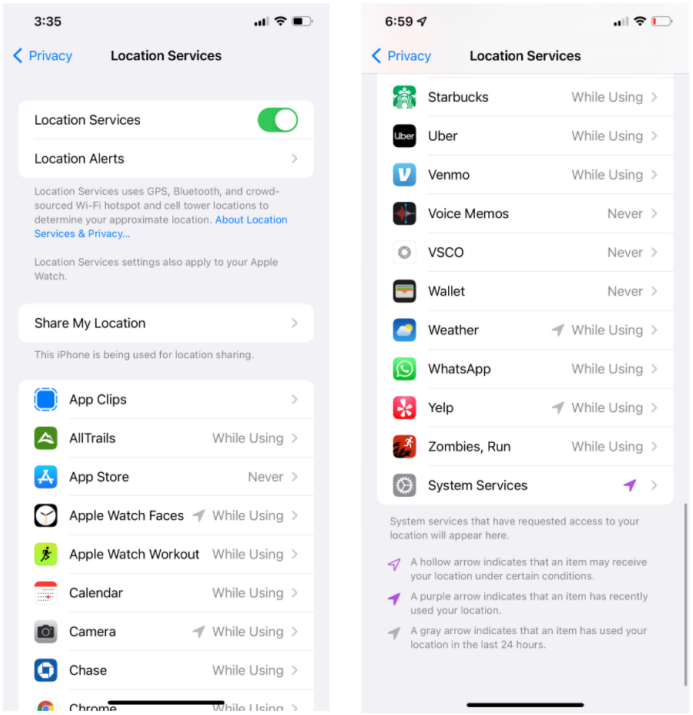

Fig. 2. Location Services page.

2.3.1 Third Party Applications

Third-party apps that use device location are shown in the first section of Location Services page. When user first opens a Location-Based Service in third-party app, location access prompt appears asking user's permission. User can choose one the the following location sharing options: Never, While Using the App, or Always. Starting with iOS 14, when prompt appears to ask user's permission to share device location, a precise location toggle is presented shown in Fig. 3 . This newly added option allows user to decide the accuracy level of location information being shared with application.

According to official documentation [8], Apple doesn't use cloaking algorithms and to increase user location privacy, it divides world map into regions and references them instead of user's precise location is user opted out of sharing their precise location. This

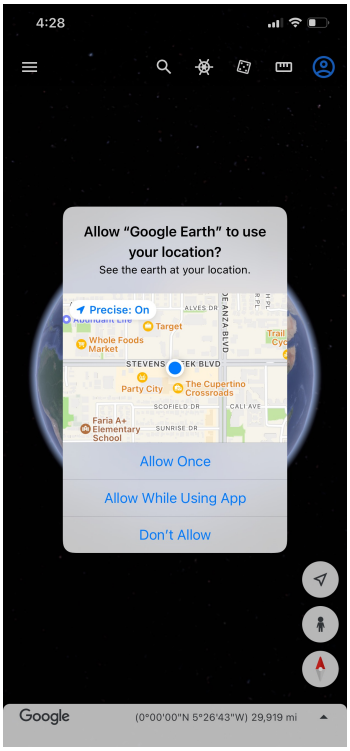

Fig. 3. Location Access Permission prompt with precise location toggle.

approach also allowed to preserve city boundaries while masking user’s precise location information. Location Privacy Framework geomasking algorithm discussed in this paper omits these restrictions.

2.3.2 System Services

System Services are using device location to provide seamless integration of personalized on-device experience. This includes services like Find My iPhone, Apple Pay, Location-Based Alerts and Suggestions, Automatic Time Zone Setting, Significant Locations, and Product Improvement services. Given user’s System Service location access permission , device can geotag all Apple Pay transactions, automatically adjust time and all time-dependent services (like alarms, calendar events, etc) based on time zone of current device location. User can keep track of all the System Services that access their location, grant/deny access anytime in Location Services settings page shown in Fig. 4.

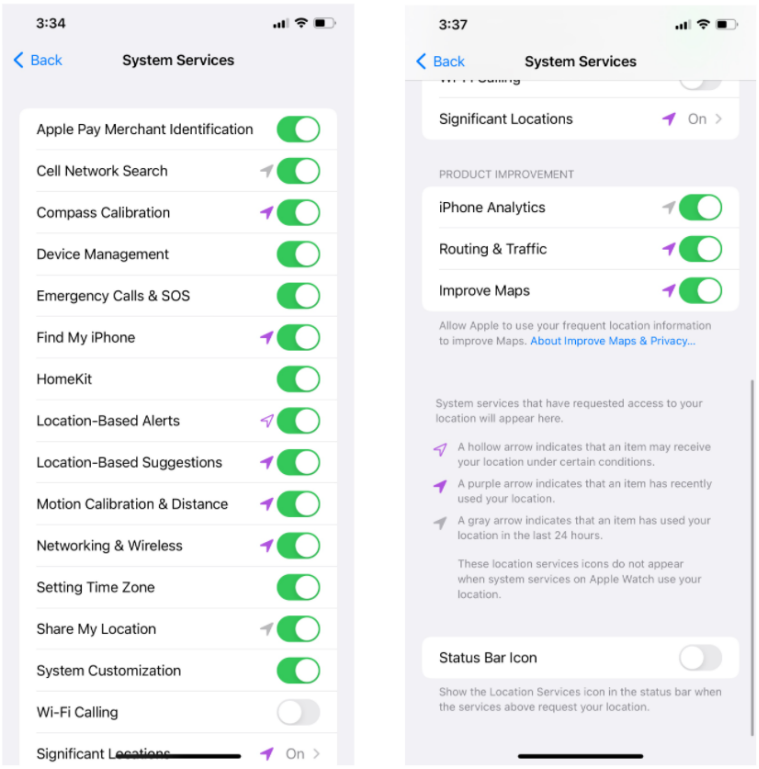

Fig. 4. System Services control page in Location Services.

Furthermore, additional information is provided whether location is shared under certain conditions (purple hollow arrow), whether location was recently shared (purple filled arrow), or if location was used in last 24 hrs (gray filled arrow), giving user additional insight about location usage by each service.

2.3.3 Product Improvement

Product Improvement location sharing control is in a separate section of System Services page (see Fig. 4). It refers to location data that is being sent to Apple for Analytics, Routing & Traffic, and Map Improvement. According to Apple, several privacy-protection mechanisms are enforced to ensure user privacy and all personal information is protected by Differential Privacy technique [9].

Differential privacy uses customized algorithms to add noise to personal-identifiable information to the whole dataset. Instead of removing all identifiable information altogether, this technique allows preserving data trends useful for Analytics without exposing identifiable data.

2.3.4 Significant Locations

Significant Locations is a personalized service that saves user mobility data on device. iOS user can access it in System Services page (see Fig. 4) and see Significant Locations page is shown in Fig. 5.

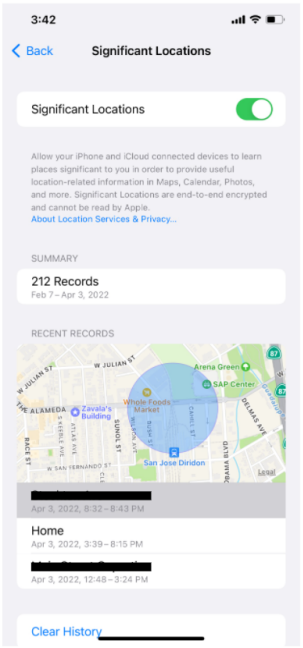

Fig. 5. Significant Locations control page.

According to Apple, Significant Locations collects information about “address the user traveled to, when they traveled there, how long they stayed, the amount of time spent commuting to the location, the method used to reach the location (e.g., by car, by walking, etc.), and the total number of times the user has visited that place” [10]. That means user’s iOS device with Significant Locations feature keeps history of all the places you

ever visited for years! However, this information is end-to-end encrypted and is not shared with Apple or any third-party applications. This data never leaves the device and is processed locally to provide users with location-based recommendations and personalized features in native Apple apps like Maps, Calendar, Photos, etc. No third-party applications can gain access to Significant Locations. Like with other location-accessing services, user can delete all Significant Locations data from device or turn it off altogether anytime through Location Services.

iOS integrated a number of location-privacy preserving techniques, such as on-device data processing for Significant Locations, Differential Privacy for collected personal data, and have taken steps towards more gradual control over location accuracy shared with user-controlled precise location sharing option. Furthermore, each iOS application submitted to App Store goes through an approval process that, among other things, checks whether device resources app is requesting adhere with Apple Store Guidelines. App developers are also required to provide publicly available privacy policy for user with resources usage justification.

However, due to how sensitive location data is, there is a need for additional control over location data management and location sharing. Instead of giving away user's precise coordinates upon request, Location Services can implement additional location privacy measures, such as providing options for sharing aggregated data related to user's location (for instance, administrative bounds, city, county, state, country) or mechanisms for cloaking location data within minimum accuracy specified by location app.

2.4 iOS: Core Location Framework

Core Location is native iOS framework that is used by developers to access device location. According to Apple documentation [11], Core Location provides three modes for location request: "standard location service" continuously sends location data, "significant-change location service" updates user location with less frequency and in

more power-efficient way for the device, and “visits location service” lets the app know if user moved from one location to another and how much time user spent there [11]. However, these modes only regulate frequencies of location updates sent from the device and are driven by the power efficiency and performance considerations rather than user privacy and accuracy levels of location data for common location functionalities.

App Store Guidelines for data use should provide detailed protocol for granular location access with recommended levels of data precision based on specified LBS functionalities. Each submitted app that goes through Apple vetting process would have to be checked for location data type requested and whether the category and accuracy of inquired information is necessary for performed location functions.

3 RELATED WORK

3.1 Location Privacy Architecture

There are two major solution design approaches in addressing location privacy in mobile devices - dummy-location generating mechanisms and trusted-third party-based.

The dummy-location-based approach does not require any additional middle parties between the device and location apps. Instead, it relies on an algorithm to anonymize user location and pass it to third-party location apps without compromising user privacy.

For example, Natgunanathan, Mehmood, Xiang, Gao, and Yu used security analysis, sensitive location cloaking, and k-anonymity computation to generate a dummy location that is sent to location apps [4]. Major downside of this approach is the computational cost overhead of complex algorithm generating dummy location, which puts a considerate strain on limited computational resources of the mobile device. Furthermore, using a generated fake location may yield incorrect or redundant responses in location apps.

Trusted third-party (TTP) solution approach introduces anonymizing party that acts as a middleman between a device and un-trusted location-based apps. TTP receives raw location data from the device, processes it, and sends anonymized data to location apps.

Gruteser and Grunwald implemented a trusted third-party solution to geoprivacy - in their approach TTP server receives user's location coordinates, anonymizes it by removing all identifying metadata, and adds perturbation noise by applying a cloaking algorithm, and then passes it to location apps [12]. However, this approach allows servers to communicate back to the app without trusted middleware, leaving an opportunity for a server response to include identifiable data that could be linked to the request and, subsequently, mobile user device. In my solution discussed in the later section, I address this shortcoming by making sure that all communication between untrusted third-party apps and servers is going through the trusted middleware.

3.2 Privacy Issues in Mobile Platforms

Recent research of privacy issues in major mobile operating systems (Android and iOS) indicates that allowing mobile apps to share users' unique data with tracking companies poses the greatest threat to personal privacy on mobile platforms. In "PiOS: Detecting Privacy Leaks in iOS Applications" [13], Egele, Kruegel, Kirda, and Vigna studied user privacy issues in iOS, and they discovered that one of the most significant issues to be the fact that unique device ID is being leaked by iOS devices for over half of the mobile apps, usually due analytics and advertising tracking embedded in apps. Furthermore, in research by Hornyack, Han, Jung, Schechter, and Wetherall it was found that Android devices also share device ID information with three prominent Analytics and Advertising companies - Flurry, Mobclix, and Greystripe [14]. Device ID is primary data that is used for aggregating data (including location data) for each user profile and presents great threats to user re-identification. While analyzing tracking destinations, Egele, Kruegel, Kirda and Vigna discovered that "82% of the applications that rely on third-party advertising libraries include AdMob" [14], which means that one company has access to data retrieved from 82% of apps on user's iPhone device. Authors suggested combining device and application data to form device ID to prevent third parties from collecting data about devices across multiple apps [14]. However, this does not prevent tracking companies from accumulating data about the user from each app individually. Especially considering the high uniqueness of human mobility patterns, location data retrieved even from one location-based app poses significant security and privacy threat to the user.

In some cases, developers are embedding tracking libraries into apps to enable monetization without being aware of how much sensitive data those libraries extract from the device. In Agarwal and Hall's application case study [15], ProtectMyPrivacy(PMP) application flagged movie-recommendation app Flixster to be accessing user's address

book data. Flixster developers contacted PMP authors claiming that the flagging was false. However, after extensive app forensics, it was uncovered that the third-party tracking library used by developers was extracting contacts, location, and demographic information from user devices - all of which were happening without Flixster developers' knowledge [15]! Discussion is needed about more control and regulation regarding apps collecting user location and usage of such data.

This allows mobile apps to extract and sell much more information about the user than it is justified by their functionality. As third-party apps are accessing location coordinates from a device, they can also extract other data that allows user profile tracking. Privacy vs utility tradeoff is currently not balanced as LBS and Advertising & Analytics data collecting companies are having disproportionate level of control due to lack of location data control from mobile operating systems. Location Privacy Framework framework works with location-based apps, device operating systems, and app servers in a system that aims to shift the balance towards privacy protection.

3.3 Mobile Location Privacy Solutions

Mobile devices should not be sharing more information with third-party apps more than it is necessary for them to perform their functionalities. Sharing more location details results in higher possibilities of user tracking and re-identification threats. Since major mobile device permission managers lack enforcement of finer control of location sharing, numerous studies have been focusing on possible granular location control solutions for interactions between mobile devices and third party LBSs. However, majority of studies remain very limited in the scope as solutions being tied to one specific location functionality.

LP-Guardian solution by Fawaz and Shin chose to provide solution for "app requires location with high granularity" category by extending existing functionality of Location object in Android core platform so that when app requests location from a device, user

has options on how they prefer location being shared [16]. K. Micinski, P. Phelps, and J.S. Foster developed CloakDruid Android tool that allowed user to determine level of location truncation of data shared with third-party Location-Based Social Network mobile apps for searching nearby lists functionality.

J. Joy, M. Le and M.Gerla suggest incorporating privacy module with granular location privacy interface into GPSD service daemon collects location information from GPS receiver [17]. This way, even system services that retrieve location data from the device would can only access a certain level of location data. However, modern operating systems are using multiple sources to determine device location. For instance, according to official Apple documentation, Core Location framework "gathers data using all available components on the device, including the Wi-Fi, GPS, Bluetooth, magnetometer, barometer, and cellular hardware" [18]. Therefore, only adding additional module to GPSD may not guarantee increase in location privacy.

However, all mentioned above solutions lack discussion on how such granular location access could be integrated into mobile platform as part of system that not just places responsibility of choosing the right level of access on the user, but also restricts LBS from requesting unnecessary location details. It could be redundant and confusing experience for user to manually specify privacy level for each LBS in order to achieve increased privacy. The default location sharing mode should not be user's full precision coordinates, but determined based on location service type.

This paper aims to provide a practical insight on how existing system (iOS) can implement this approach addressing location privacy under the assumption that if users agree to share their location, they would prefer to have as much privacy as possible while still being able to use location app functionalities.

4 PROJECT DESCRIPTION

4.1 System Design

Despite the high sensitivity of location data, whenever an app needs any location information from the user, it is able to get exact latitude and longitude coordinates of the device, regardless of location accuracy it needs to perform its functions. This allows mobile apps to extract and sell much more information about the user than is justified by their functionality. As third-party apps are accessing location coordinates from a device, they can also extract other data (shopping preferences, photos, contacts, demographics) that can be linked to user's location in profile tracking. Starting with iOS 14, Apple gave user ability to disable precise location sharing, but it did not prevent location apps to request more location details that it is necessary for their functionality. Currently, it is up to the user to decide whether to share approximate or precise location data with each app.

In my solution design, I introduced a trusted third party Location Privacy Framework. The framework works with location-based apps, mobile device operating system, and app servers in a system that provides user more location sharing options while preserving location app functionalities (see Fig. 6). LPF addresses two main location privacy concerns in the system by limiting precision of the location data extracted from a device and controlling the way it is shared with third party servers.

4.1.1 Mobile Operating System (iOS)

It is the mobile platform's job to make sure that applications do not request more information than they need for providing their functionality, as each platform has its own verification process that each application has to go through to be approved and posted in the application store for users to download. For instance, Android has open-access procedure that lets users know about how much mobile data applications are accessing. Apple follows vetting process to ensure that app developers follow the guidelines stated in the developers' license agreement and because Apple doesn't explicitly set up the

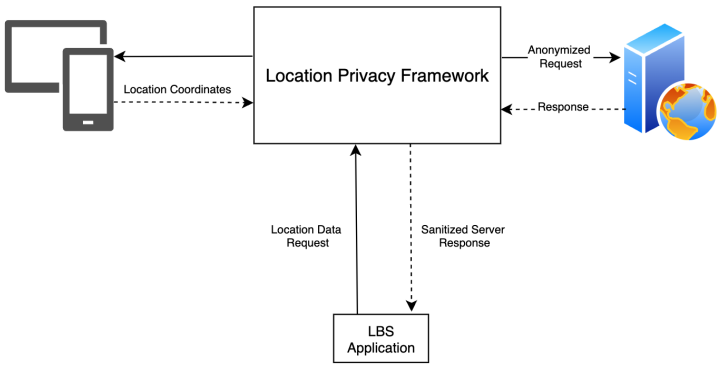

Fig. 6. System Design Using LPF.

technical limit of application access to the operating system, there have been documented cases of malicious apps passing Apple's verification process (probably due to human mistake), putting user privacy and security at risk.

Location Privacy Framework is intended to work alongside mobile operating systems to gain more control over how much information about the device and its location is being collected by third-party applications. Apps don't have access to the mobile device directly, and they are not allowed to make any first-hand network calls to third-party servers, which prevents third parties from tracking user profiles and gaining access to any information that might be used for identification. Hence, any type of location data collected from the user device in this system is protected and anonymized based on the minimum accuracy level needed for app functionality.

4.1.2 Location Privacy Framework

All third-party location data transmission must go through the LPF, ensuring that apps are only getting the data they requested and information sent back-and-forth between servers, and the application does not enable user tracking.

Furthermore, the third-party location-based application must include specifications of the location accuracy level that it needs to perform its functionalities in the request to the framework. As part of request to obtain any location-related data from the device, LPF enforces apps to specify the category of request and minimum accuracy required for app functionality. Based on this information, LPF applies aggregation or geomasking and then sends anonymized data to remote server, processes the response, and returns it to the application. If LPF discovers that user is located in highly identifiable area during anonymization step, it sends out multiple dummy requests to ensure k-anonymity for the data third-party server receives.

4.1.3 Third-Party Location Applications

Applications that interact with LPF do not retrieve user location information directly from the device; they only get the location-related data that the server returns as a response to a request from LPF. When location app feature needs user location, it makes a request to Location Privacy Framework with the following data: type and accuracy of location data, server destination, expected parameters, expected response format. LPF returns processed server response to the app, so it can perform its location-based functionality.

Applications should not be able to send any location request network calls bypassing Location Privacy Network, making it more difficult for unauthorized third-party data brokers to obtain any data that could be used for user profiling. Even though these restrictions may limit app's functionality scope, they are enforced by the system to enhance security, protect user's privacy and control personal data collection by third-party applications. This way, apps cannot track user's mobility trace even locally on the device. This would limit the damage of personal data breaches in case malicious applications are installed in the device.

4.1.4 Third-Party Location Servers

A third-party server is the only untrusted member of the system that is capable of retrieving location data that can be used for user profiling. To reduce the chance of tracking, Location Privacy Network generalizes the data to the minimal accuracy level (that is verified and enforced by the mobile operating system during the application submission process to the mobile platform store) and sanitizes it from all additional elements. Hence, the server only receives anonymized location data without any identifying information that may link it to a particular device.

4.2 Anonymization Algorithm

4.2.1 Assumptions

Adaptive geomasking algorithm uses population density data to achieve k-anonymity for user location request, so Location Privacy mechanism assumes that system have access to this information, whether it is pre-installed or retrieved. Location Privacy Framework implementation discussed in Chapter 5 uses small previously generated population distribution dataset to simulate real-world environment. However, how population density data is accessed real-time based on user location is not addressed by this paper.

4.2.2 Algorithm Description

Given Location-Based Service(LBS) location request with minimum accuracy A and system-defined crates for k-anonimity and t-closeness, after LPF retrieves device coordinates, it proceeds with anonymization step. Anonymization function takes in device's real location coordinates(*LAT* and *LONG*) and returns AnonymizedLocations and is presented in Algorithm 1 that summarizes the following steps.

If computed radius R is smaller than t-closeness threshold, R takes value of t. Initial radius R computation only ensures that there are at least k-1 users included in the anonymized area. However, if user is located in very high-density region, initially

Algorithm 1 Anonymization Algorithm

```
1:  procedure ANONYMIZE(LONG, LAT)  ▷ Returns K locations that anonymize Longitude and Latitude
        // Define K, R_min, R_max And AnonymizedLocations to save the result
2:      K ← 10
3:      R_min ← 10.0
4:      R_max ← 100.0
5:      AnonymizedLocations ← [ ]
6:
        // Get nearest K users from Long and Lat
7:      KNearstUsers ← GETKNEARSTUSERS()
8:
        // Get the furtherest user from Long and Lat in the list KNearstUsers
9:      MaxDistanceToAKUser ← GETMAXDISTANCEFROMUSERS(KNearstUsers)
10:
11:     if MaxDistanceToAKUser < R_min then:
12:         MaxDistanceToAKUser ← R_min
13:         AnonymizedLocations ← KNearstUsers
14:     else if MaxDistanceToAKUser > R_max then:
15:         MaxDistanceToAKUser ← R_max
16:         for i = 1...K do:
17:             AnonymizedLocations[i] ← GENERATERANDOMLOCATIONWITHINRANGE(R_min, R_max)
18:     else:
19:         AnonymizedLocations[i] ← KNearstUsers
20:     return AnonymizedLocations                    ▷ List of anonymized locations with size K
```

computed R would be very small and could reveal too many details about user's location. To account for cases like this, algorithm enforces additional t-closeness constraint, safeguarding from geomasked value being too close to original location.

If computed radius R is larger than maximum LBS accuracy A, anonymization area takes value of A. If user is located in very low-density region, there may not be k-1 people within the accuracy radius limit defined by LBS. This case needs more elaborate solution that would keep user location anonymized and preserve the accuracy that LBS needs. To achieve that, algorithm applies donut geomasking method with radius R=A (max LBS accuracy) to anonymize not only user's original location, but its k-1 closest neighbors as well. K geomasked points are then used by LPF to generate k-1 extraneous dummy requests together with one valid one. This approach preserves user's k-anonymity,

since server cannot distinguish true request out of k LPF sent. Since each request used location point geomasked within accuracy limit A, LBS functionality is preserved.

If Location Privacy Framework to be implemented as part of native iOS location handling framework, it would be able to access Significant Location feature with user's spatiotemporal history data. This would give the framework capacity to analyze user's behaviors and come up with more plausible dummy location generation mechanism. All latest iOS devices come equipped with Neural Processing Unit hardware called Apple Neural Engine optimized for on-device machine learning computing. Given access to user's location history, LPF could potentially apply machine learning to learn patterns about user's schedule and use them for location generation that is realistic in the context of daily habits of a specific user.

If computed radius is within the boundaries of t and A, system proceeds with computed radius R to geomask the location point in donut-shaped anonymization area. Resulting algorithm ensures k-anonymity and t-closeness anonymization criteria, while preserving accuracy constraint necessary for LBS functionality. It is important to point out that this anonymization implementation supports one-time location request by LBS, not taking into account functionalities that involve tracking multiple requests (such as navigation, routing). Furthermore, geomasked points are generated randomly with anonymized donut-shaped area without considering geographical object locations, such as water bodies, pedestrian sidewalks or road networks. These considerations could be included in future algorithm versions.

4.2.3 Anonymization Algorithm Demo Application

Screenshots in Fig. 7 demonstrate donut-shaped anonymization areas from Location Privacy Framework Anonymization Algorithm applied to multiple location points for different precision constraints set by LBS, and k-anonymity values set by LPF.

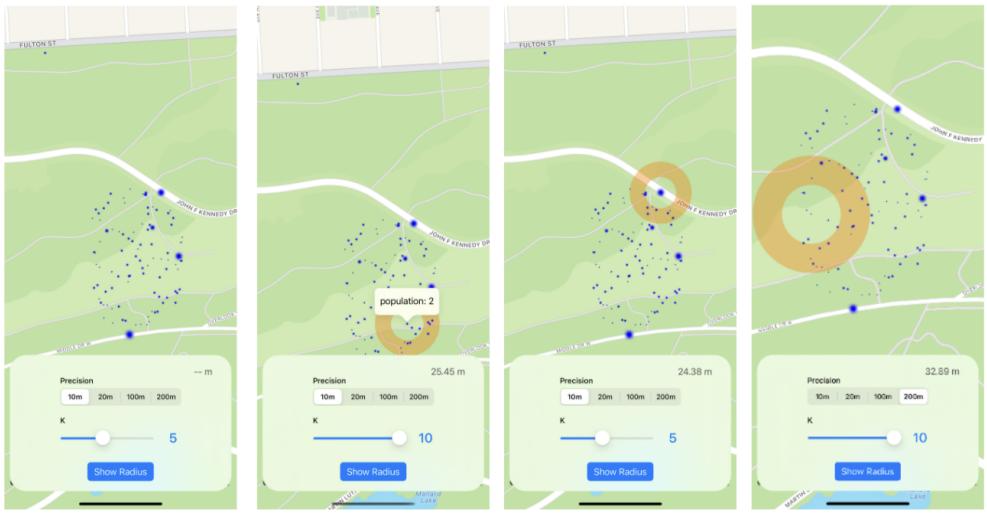

Fig. 7. Anonymization Area for different location points.

4.3 Threat Modeling

The main goal of the solution system is to minimize amount of identifiable location details shared with third-party applications, servers, and any other type of adversaries that may gain access to that information. Adversary can possess knowledge derived from publicly available sources, geographical region characteristics, etc., that can be used to gather background knowledge about user's locations. They are interested in gathering as much location data as possible in user profile to make capital of it by providing additional personalized services or recommendations. Furthermore, more details in user profile increase the value of information in data broker market. Therefore, solution system prioritizes on-device data anonymization processing, so that any location data shared with location app or server is as anonymized as possible.

LBS server that receives Location Privacy Framework request is central point for adversary modeling, because it's an untrusted party that potentially receives the most location details from multiple requests. LPF anonymization algorithm makes sure that

location data server receives is k-anonymous with t-closeness measure to address Homogeneity Attack risks.

To prevent user subsequent location data tracking by LBS server, LPF need to take additional measures to make it harder for the server to trace the source of each request. That involves data sanitization step, that removes device identifiers and any other unnecessary characteristics that may be used for profiling from the request data. If location data is very sensitive, Location Privacy Framework can adopt additional privacy measure, such as creating a new session for every request - making it difficult to link group multiple requests to the same device. These measures would further increase time overhead, but provide more protection against profiling.

Location app collects less location details that LBS server, since its main goal is to display results of LBS server location query, and allow user to interact with it. All network communication containing location data must go through LPF, so the system does not allow location app to share location data with any third parties. Location apps can persist location-related data, but since system have control over app communications, adversaries have less chances of getting their hands on it. Furthermore, in order for iOS application to published in Apple Store, it must abide with App Store Review Guidelines. However, Apple review process is not perfect and there are known cases of malicious apps finding their way into App Store that abuse user personal data.

Solution system provides possible improvements in addressing current privacy threats in mobile platforms. Proposed mechanism takes into account the nature of location data and types of mobile location functionalities to ensure effective anonymization. It is also designed to address most privacy-threatening components of current iOS environment to adjust unbalanced location privacy-utility tradeoff.

4.4 Use Cases

Location functionalists considered in Location Privacy Framework design are categorized into two groups by required location data type.

First type of location services rely on a specific location attribute to perform its functionality. These services do not need to know the accuracy of the location point, they just need to know which characteristic group user's location coordinates fall into, mapping it into an aggregated point. A very common example is the weather app because it only needs the city of user's current location to perform its functionality and show accurate weather information.

Fig. 8 diagram shows the details of how a weather app would operate in proposed system. When weather app needs location data, it sends a request to LPF with server destination, parameters it needs, expected response, and specifying that in order to get server response with needed weather data, it would require city-level aggregation information. LPF extracts location coordinates from the device and aggregates location data to city level. LPF sends a request to LBS server with parameters weather app provided, making sure that location data is anonymized and do not contain any identifiers that third parties can use for tracking. Once LPF receives expected weather data, it sanitizes it and passes it to weather app. Now weather app can display relevant weather information for user's current city.

Second type of LBS need to have the coordinates of user geolocation within specified accuracy bounds. For example, nearby locations locating app like Yelp needs to send their server location coordinates within some accuracy bounds to receive relevant response with points-of-interests nearby matching user's query.

Fig. 9 diagram showcasing Yelp operating in LPF system. It operates similarly to a weather app from previous example, except LBS server needs actual location coordinates to return points of interest nearby. Hence, LPF applies Anonymization Algorithm

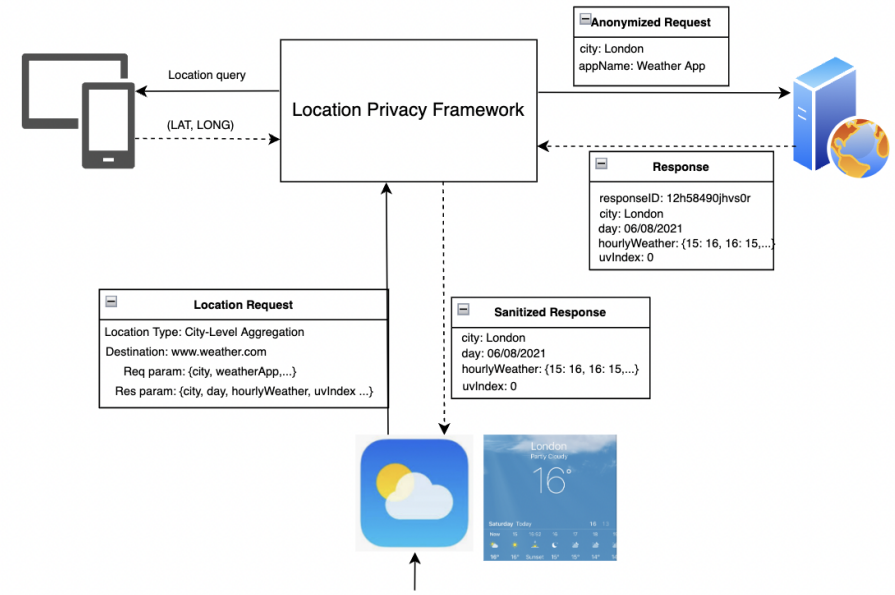

Fig. 8. Diagram for showcasing weather app functionality using LPF.

described in Algorithm Table to location attributes retrieved from the device and passes them to LBS server.

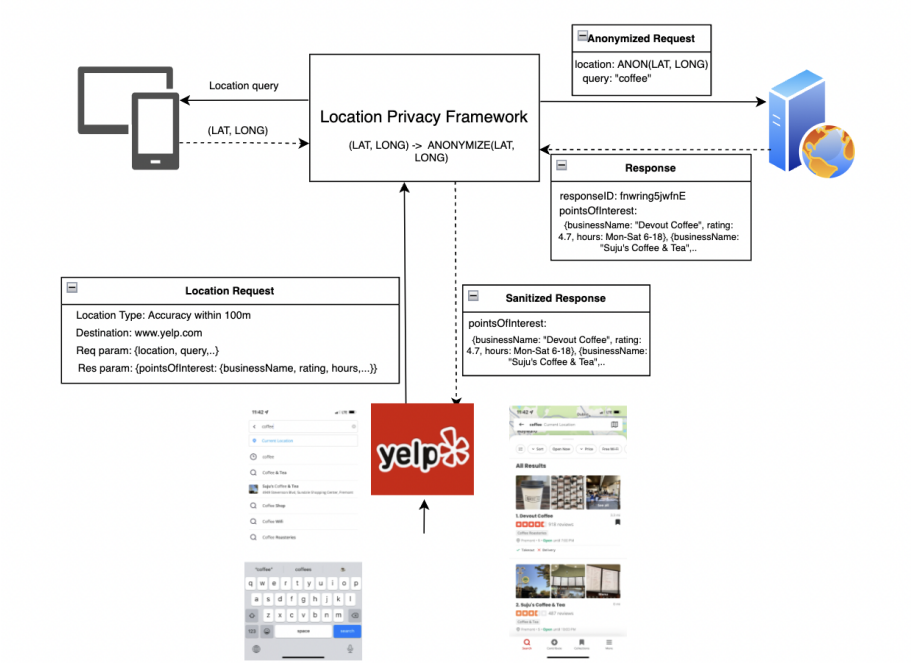

Fig. 9. Diagram for showcasing Yelp app functionality using LPF.

5 IMPLEMENTATION AND RESULTS

5.1 Technology Used

The application was developed using Xcode Version 13.3 and Swift 5. It uses UIKit framework for UI implementation, CoreLocation framework to get current device location, and Foundation framework to make network calls using URLSessions. The backend server is hosted on Heroku and it uses Flask to implement API endpoints. The backend consists of one POST API /location, which returns whatever location it received in the request. The purpose of the API is to demonstrate the final values the server receives for the longitude and latitude (since the request is compiled, sent and handled by LPF, it's good to demonstrate what eventually makes it to the server, and how the server response is handled).

5.2 User Interface

The application User Interface is very simple: there is a toggle to turn on/off location anonymization in the top section of the screen, and data section with labels at the lower section of the screen. Data Section shows device's real longitude and latitude coordinates and longitude and latitude coordinates that are sent to server. I have created a simple backend server that takes in longitude and latitude coordinates and return the exact same longitude and latitude coordinates in the response for demo purposes. Using this, the location privacy framework provides an interface for developers to pass data that they need to send to their own server, and the type of location they need access to, and the location privacy framework will inject their requirement to the REST request before sending it. Attached in Fig. 10 is an example where an app is using the Location Privacy Framework to create a POST request with url and request body with location (long and lat), and then the request is sent by LPF and a JSON response is send back to the app.

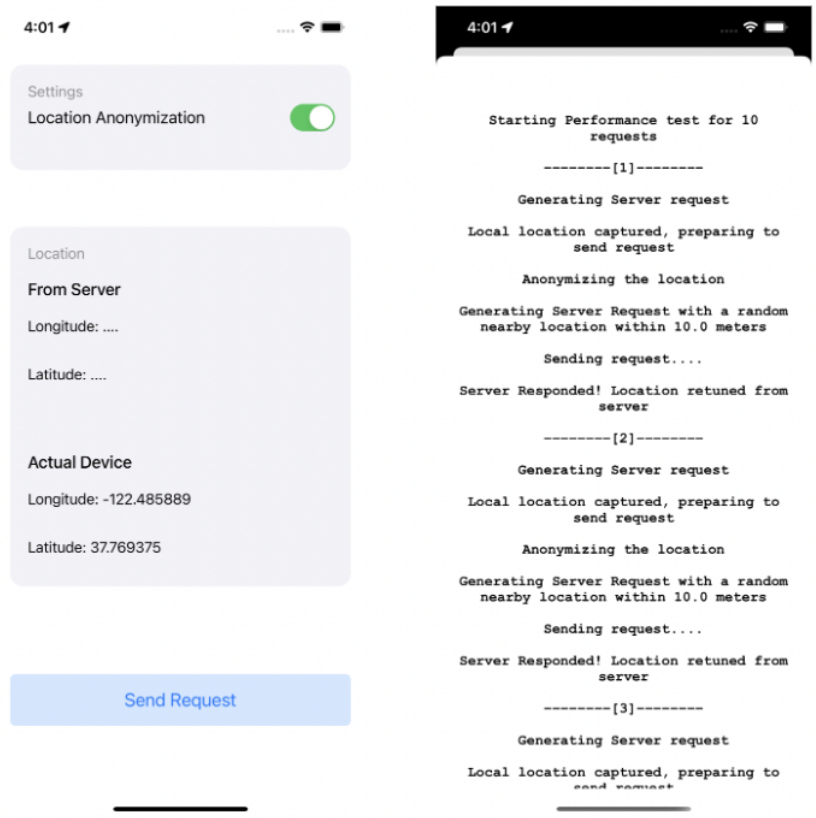

Fig. 10. User Interface of Location-Based Application using LPF.

5.3 Implementation

The framework uses a factory-like design, where the developer can only create an instance of the LocationPrivacyRequest by calling the create method in Fig. 11, in which they needs to specify the server url, the type of REST request, and the parameters needed to be sent to the server.

Once the request is created, and the developer decided that it's the right time to send the request (Fig. 12), they can call the send method on the instance and the LPF will

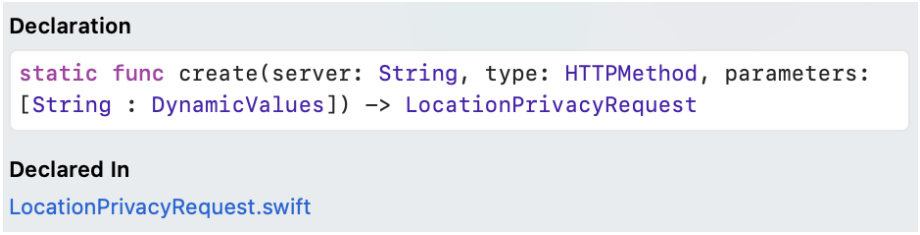

Fig. 11. The prototype for the create function, it takes in server, request type and returns an instances of LocationPrivacyRequest.

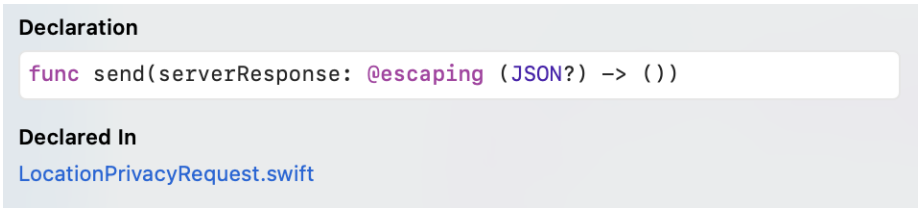

Fig. 12. The prototype for the send function, with a server response callback.

asynchronously send the request and return the response back in a JSON format using the callback pattern.

5.4 Results

Table 1 introduces time results of LBS sending 10 location requests with and without LPF anonymization. Anonymized requests result in insignificant overhead 0.01 ms per request on average. Last row represents an extreme case when there is no other people around user's location within location accuracy boundaries, so LPF has to generate 9 fake locations for each LBS request, resulting into 100 requests sent by LPF per 10 LBS requests. This resulted in large overall time delay for LBS of over 7 ms, but on average each LPF request was faster (0.09 ms vs 0.13 ms for non-anonymized request). This is because for each 10 LPF requests, device location was retrieved only for one of them, the rest of locations used in 9 LPF requests are generated.

Table 1
Times for 10 LBS Requests

LBS Req Type	Total Duration (ms)	Num LPF Req(s)	Avg Req Time(ms)
Non-Anonymized	1.3156311511	10	0.13156311512
Case 1	1.468986988	10	0.1468986988
Case 2	9.099128246	100	0.09099128246

5.4.1 K-Value

K-value should be a topic of location functionality-specific discussion. Higher k provides higher privacy level, but more computation and longer the LBS request times. Some location functionalities require higher precision, hence to maintain user privacy, higher k-values could be suggested. On the other hand, time sensitivity of each location functionality should be considered in determining optimal value of k.

The following table Table 2 and diagram Fig. 13 shows data for the times for LBS requests for different k-anonymity criteria for low-density area where device is located in the area that has no other users within accuracy criteria. In this case, LPF anonymization algorithm generates k-1 dummy requests by applying donut geomasking to k-1 nearest points. Table 2 shows the relation between time and k-anonymity.

Table 2
LBS Request Times Based on K

k Value	LBS Req Time (ms)
1	0.22566769123077393
2	0.09888501167297363
3	0.09862709045410156
4	0.0967982530593872
5	0.12481515407562256
6	0.10080621242523194
7	0.09911746978759765
8	0.18499329090118408
9	0.27503581047058107
10	0.35946040153503417

Privacy system decides on optimal value for k based on the type of LBS and population distribution, accuracy required, time sensitivity of the service, etc. For

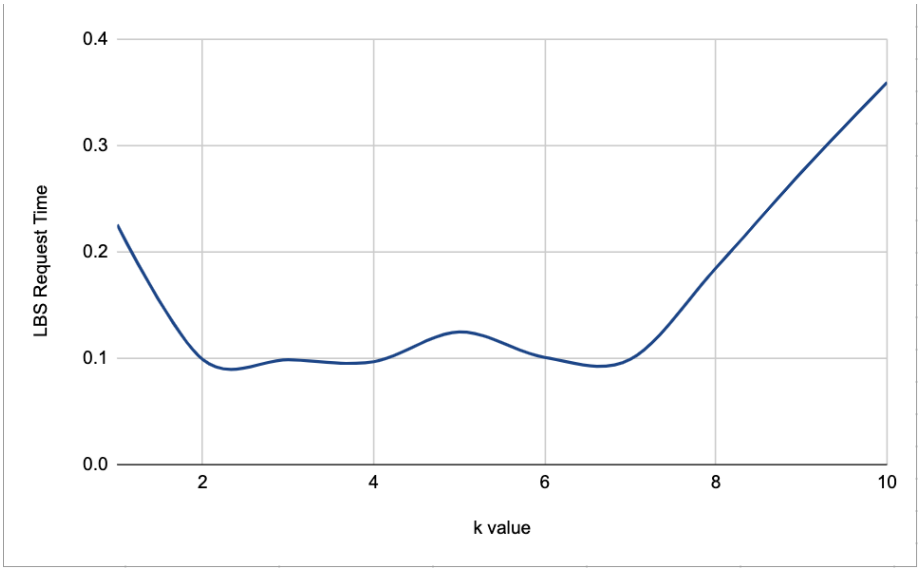

Fig. 13. Graph for LBS Request time and k-value tradeoff.

effective privacy protection in real environment, these factors should be included in uniform location privacy protocol

5.5 Integrating Location Privacy System

Location Privacy Framework is designed to be a trusted party, since it has access to device resources and is responsible for the anonymization step. Built with user location privacy in mind, solution discussed in this paper involves changes in way location data is anonymized as well the changes in the way applications installed on device communicate with third party serves. Hence, to incorporate LPF system into current iOS environment, a number of operating system functionalities and frameworks need to adjust location data handling and networking mechanism.

Apple framework responsible for data handling, Core Location, can perform LPF anonymization algorithm, returning processed location data based on accuracy level application requested. Since location data can only be communicated through LPF, changes are required in iOS network framework that handles network requests (such as URLSession) to reroute data transfer through LPF.

Furthermore, system is effective in restricting location details only to the level that necessary for LBS functionality if request location details are verified in submission review process by Apple for every third-party app in App Store. Mobile platform need to review the way applications are requesting location data, and update App Store Review Guidelines with more definite location privacy rules than "use Location services in your app only when it is directly relevant to the features and services provided by the app" [19]. In order to achieve that, more work is required towards universal convention that clearly defines and structures all existing location functionalities with corresponding detailed location privacy protocols assigned and enforced throughout the platform. This approach increase users trust and provides clear location privacy guidelines for developers to work with.

6 CONCLUSION

Throughout the research, it was discovered that most mobile platforms only support binary location access for location apps. Since a large number of mobile activities are geotagged, location data can reveal alarming amounts of personal life details, such as medical or financial information. Because the existing guidelines for location data usage defined by mobile platforms are vague, third party applications can freely collect high-precision location data from user device upon user's consent to share location. Most of location apps share personal data they retrieve from user with other third parties (such as advertising and analytics services) and personal data often end up in the hands of dominant data tracking companies. A number of studies pointed out that human mobility trace is highly unique, and can easily be used to re-identify a person with only a few spatiotemporal points. Once user is identified, data collection companies can track them and even profile them across multiple applications, accumulating alarming amount of personal data. Considering the high uniqueness and sensitivity of location data, this indicates an urgent need for new practices and regulations that would protect user location privacy and enforce secure location data management.

When iOS first came out, the only apps that were able to be installed and able to access device resources were native Apple apps until Apple introduced App Store in iOS 2.0 that allowed developers to develop applications in iOS and access device resources. Throughout the years, Apple kept adding more and more restrictions on third-party applications to limit third-party apps abusing device resources, but technology was always growing faster than privacy guidelines, especially in regards to location data privacy. Instead of incremental changes to improve information privacy on the phone, this paper proposes more drastic OS-level solution, where operating system treats third-party applications with more skepticism by enforcing additional levels of control over resource access.

This paper proposed a Location Privacy Framework as part of solution that allows operating systems to gain more oversight over the way third-party apps extract and handle location data. To protect user location privacy, granular location sharing approach is proposed. LPF considers common location functionalities in anonymization step that incorporates level-based approach in sharing user location details by adding accuracy-based anonymization and aggregation before sharing data with LBS server. LPF collects location data from the device, sanitizes the data from any identifiable information, and anonymizes data in accordance with the requested location type instead of giving LBS direct access to device location. All data aggregation is performed close to the source (locally on device) to minimize amount of sensitive location data that ever leaves the device.

Furthermore, LPF restricts network communication from third-party location apps installed on device. After anonymization is done, the framework sends data to the app server and records the server response. Because server response may attempt to send some data that may be used for the location app to track the request source, LPF also sanitizes server response before sending it back to the location app. All location data-related communication have to go though LPF to lower risks of location tracking.

The primary goal of this paper is to facilitate the conversation about location privacy in mobile devices. Location privacy safeguarding mechanism should operate under assumption that third-party apps aim to obtain as much information as possible, because more collected details increase the price at which collected data can be sold in data market. Proposed privacy-preserving approach can also be extended to include other types of data, such as different media types, user contact information, passwords, and other personal data.

7 FUTURE WORK

First major security risk of trusted-third party solution design is the risk of unauthorized parties gaining access to the framework, compromising user's data privacy. However, as discussed in section 5.5, solution was designed with the thought of how existing location handling framework can adopt more location privacy-preserving measures. In order for measures described in this paper to be effective, they should be integrated into mobile platform, causing drastic changes to the entire system.

Another limitation of the system is that a server may be able to trace back the source of the request, revealing information about Location Privacy Framework and, consequently, the device. A future proposal to lower the risk of third-parties linking location data to a specific device or profile through LPF middleware could incorporate the additional security measures. Dummy-location generation algorithm could be further improved by gaining access to the resources that only currently available to native applications, such as Significant Locations to make sure most sensitive locations are not accidentally revealed. Furthermore, if LPF can have access to pre-installed map data with population density and geographical objects, dummy-generation algorithm should be improved to consider these constraints. Improving LPF fake requests to appear more realistic should make adversaries to have harder time distinguishing real request, even with background information knowledge.

Emphasis on on-device data processing is the most secure practice, but it comes with computational overhead limitations. Since the framework resides on a device with very limited computational power, this approach would require a lot of optimizing because additional steps and requests put a significant computational strain on mobile devices and hinder location app functionality. Additional research is needed to see how can anonymization computation can be optimized for existing hardware or for new advances

in hardware like secure preprocessors can be introduced to directly address computational cost of privacy and security.

Finally, in order to establish and enforce better location-preserving practice, additional work is needed towards uniform protocol that takes into account technical needs of LBS market and nature of location data to create new standards of granular location access. Having such contract in Privacy Guidelines would provide more transparency for users on why each app needs location access of a certain level and how the data is being used.

Literature Cited

[1] "A Day In The life Of Your Data," Apple Inc., White Paper, Apr. 2021, Accessed: Apr. 5, 2022. [Online]. Available: https://www.apple.com/privacy/docs/A_Day_in_the_Life_of_Your_Data.pdf

[2] R. Binns, U. Lyngs, M. Van Kleek, J. Zhao, T. Libert, and N. Shadbolt, "Third party tracking in the mobile ecosystem," in *Proc. of the 10th ACM Conf. on Web Science*, ser. WebSci '18. Association for Computing Machinery, Apr. 2018, pp. 23 – 31. [Online]. Available: https://doi.org/10.1145/3201064.3201089

[3] Y.-A. Montjoye, C. Hidalgo, M. Verleysen, and V. Blondel, "Unique in the crowd: The privacy bounds of human mobility," *Scientific Reports*, vol. 3, pp. 3–4, Mar. 2013, doi: 10.1038/srep01376.

[4] I. Natgunanathan, A. Mehmood, Y. Xiang, L. Gao, and S. Yu, "Location privacy protection in smart health care system," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3055–3069, 2018, doi: 10.1109/JIOT.2018.2878917.

[5] L. Sweeney, "Weaving technology and policy together to maintain confidentiality," *The Journal of Law, Medicine Ethics*, vol. 25, no. 2-3, pp. 98–110, Jun. 2007, doi: 10.1111/j.1748-720X.1997.tb01885.x.

[6] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," SRI Int., Oakland, CA, Tech. Rep. SRI-CSL-98-04, May 1998.

[7] K. H. Hampton, M. K. Fitch, W. B. Allshouse, I. A. Doherty, D. C. Gesink, P. A. Leone, M. L. Serre, and W. C. Miller, "Mapping health data: Improved privacy protection with donut method geomasking," *American Journal of Epidemiology*, vol. 172, no. 9, pp. 1062–1069, Nov. 2010, doi: 10.1093/aje/kwq248.

[8] N. Patrick, "Understanding approximate location permissions in ios 14," Radar Blog, Jul. 2020, Accessed: Apr. 5, 2022. [Online]. Available: https://radar.com/blog/understanding-approximate-location-in-ios-14

[9] "You Have Control Over What You Share," Apple Inc., Accessed: Apr. 5, 2022. [Online]. Available: https://www.apple.com/privacy/control/

[10] "Privacy Overview," Apple Inc., White Paper, Nov. 2019, Accessed: Apr. 5, 2022. [Online]. Available: https://images.apple.com/privacy/docs/Location_Services_White_Paper_Nov_2019.pdf

[11] "Getting the User's Location," Apple Inc., Developer Apple Documentation, Accessed: Apr. 5, 2022. [Online]. Available: https://developer.apple.com/documentation/corelocation/getting_the_user_s_location

[12] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proc. 1st Int. Conf. Mobile Systems, Applications and Services*, ser. MobiSys '03. New York, NY, USA: Association for Computing Machinery, Jan. 2003, pp. 31–42, doi: 10.1145/1066116.1189037.

[13] M. Egele, C. Kruegel, E. Kirda, and G. Vigna, "PiOS: Detecting Privacy Leaks in iOS Applications," in *Proc. Network and Dist. Syst. Security Symp. (NDSS)*, Jan. 2011.

[14] P. Hornyack, S. Han, J. Jung, S. Schechter, and D. Wetherall, "These aren't the droids you're looking for: Retrofitting android to protect data from imperious applications," in *Proc. 18th ACM Conf. on Comp. and Communications Security*, ser. CCS '11. New York, NY, USA: Association for Computing Machinery, 2011, pp. 639–652, doi: 10.1145/2046707.2046780.

[15] Y. Agarwal and M. Hall, "Protectmyprivacy: Detecting and mitigating privacy leaks on ios devices using crowdsourcing," in *Proc. of the 11th Annu. Int. Conf. on Mobile Systems, Applications, and Services*, ser. MobiSys '13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 97–110, doi: 10.1145/2462456.2464460.

[16] K. Fawaz and K. G. Shin, "Location privacy protection for smartphone users," in *Proc. 2014 ACM SIGSAC Conf. Computer and Communications Security*, ser. CCS '14. New York, NY, USA: Association for Computing Machinery, 2014, pp. 239–250, doi: 10.1145/2660267.2660270.

[17] J. Joy, M. Le, and M. Gerla, "Locationsafe: Granular location privacy for iot devices," in *Proc. Eighth Wireless of the Students, by the Students, and for the Students Workshop*, ser. S3. New York, NY, USA: Association for Computing Machinery, 2016, pp. 39–41, doi: 10.1145/2987354.2987365.

[18] "Core Location Framework," Apple Inc., Developer Apple Documentation, Accessed: Apr. 5, 2022. [Online]. Available: https://developer.apple.com/documentation/corelocation#//apple_ref/doc/uid/TP40007123

[19] "App Store Review Guidelines," Apple Inc., Mar. 2022, Accessed: Apr. 5, 2022. [Online]. Available: https://developer.apple.com/app-store/review/guidelines/#privacy