Development of Cybersecurity Lab Exercises for Mobile Health

Hongmei Chi
hongmei.chi@famu.edu

Ashok Srinivasan and Meysam Ghaffari
asrinivasan@uwf.edu sg15w@my.fsu.edu

Institution
Address

Abstract - There is an emerging class of public health applications where non-health data from mobile apps, such as social media data, are used in subsequent models that identify threats to public health. On one hand, these models require accurate data, which would have immense impact on public health. On the other hand, results from these models could compromise privacy of an individual's health status even without directly using health data. In addition, privacy could also be affected if systems hosting these models are compromised through security breaches. Students ought to be trained in evaluating the effectiveness of different protocols in ensuring privacy while providing useful data to the models.

There is a lacuna in current cybersecurity education in training students in the context of both the above types mobile health applications. The objective of this paper is to develop educational material to augment current cybersecurity courses for undergraduate and graduate students. We will develop material to teach about fundamental issues related to security and privacy in mobile health applications, and produce a cloud-based hands-on lab that lets students explore consequences of different solution strategies. Lab exercises will provide students with insight into development of practical solutions based on sound theoretical foundations.

Keywords

Hands-on lab, mHealth app, privacy and security, health data, practical solutions, K-anonymity

1. INTRODUCTION

Mobile devices, such as cell phones and tablets, are a convenient means of managing health care and an effective means of promoting healthy behavior, because they are popular, and most users carry them with themselves everywhere [1]. For example, in 2017, 95% of US adults had a cell phone and 77% of those phones were smart phones. While a survey in 2012 showed only 31% of users using healthcare apps, recent surveys shows 62% of cell phone holders using at least one healthcare app [2]. This is a substantial growth in the number of users of such services.

Mobile health applications have been found to facilitate earlier and more effective intervention of both physical and mental illnesses. This has led to most US smartphone users installing at least one health application on their phones. There is usually exchange of data between such applications and servers on the cloud that store and process much of the information. Security of health data and systems hosting them is clearly important. Cybersecurity education ought to train the workforce on techniques for protecting such information and systems.

While the above deals with individual health, there is also an emerging class of public health applications that use non-heath data to predict consequences to public health. For example, a recent study on using mobile devices for modeling disease outbreaks after an earthquake used cell phone location data to determine human movement patterns, which proved to be more accurate than government data [3]. Google Flu Trends and Google Dengue Trends used search query results to try to predict disease outbreaks. Google Health Trends continues making new data available to researchers. There is also much current research that uses social media data to identify health status for the purpose of public health [6]. Much of the social media data and search query data are generated through mobile devices. Models could use non-health information, gleaned from these sources, to identify likelihood of diseases in an individual. Due to these sophisticated scientific models, security vulnerabilities of otherwise innocuous data could be used to infer health

information, thereby compromising the privacy of individuals. These vulnerabilities could arise either in the transmission of the information or in the cloud-based systems running the models.

Privacy is an important issue in these mobile healthcare applications, because users are concerned about their health information being revealed. Consequently, much research has been performed on preserving privacy while providing a user with mobile health features [4]. For example, anonymization can be performed using a trusted anonymization server or other methods such as P2P anonymization [5]. With the increasing concern for safety and integrity of information against security data leakage, it has become mandatory that organizations follow strict guidelines and security framework to assure the safety and protection of data and systems [2]. In addition, protocols can be designed so that privacy is not compromised.

Currently, there is a lack of education material that trains students in security and privacy solutions arising from mobile health applications. Given the rapid developments in this field, students with training in this field would have good employment opportunities. We will address this lacuna in cybersecurity education by developing tutorial material on the security and privacy issues in mobile health contexts, along with a set of exercises deployed on a cloud-based lab, that will help students explore different security and privacy vulnerability scenarios. This could have a transformative effect on cybersecurity education in Florida.

2. DESIGN

Communication from a device to its destination can be secured using an encrypted protocol. However, this is not adequate when a lot of information is sent, such as location, health information, etc, because identity and health status could be inferred. Ideally, even the server running the model should not infer identity, and the results of the model should not reveal this either. Several protocols have been developed to try to attain this objective, including by graduate students in the proposed research team [5].

The simplest method is Pseudonym where instead of ID, users use a pseudonym that could be changed periodically. The problem of this method is that it has a communication overhead when it wants to change the Pseudonym. Moreover, it's possible to link different Pseudonyms to each other based on their behavior pattern. Furthermore, it does not preserve privacy in sparsely populated locations.

Sending fake queries is another approach to just hide the real queries among the fake ones. The biggest issue with this method is that it will cause a huge overhead over the server. Also using developed techniques like map matching lots of fake queries could be ignored easily.

The Perturbation method tries to achieve privacy by adding noise to the queries, such that the exact location of the user could not be detected. This approach is good since it could be easily deployed without needing anything specific. But the problem is that since we don't know how much noise is needed to hide the identity of the user, sometimes even adding noise is not going to work. Besides, since we don't know the optimum needed noise to hide the identity of the user, we will have a significant quality loss since the queries are not precise anymore and thus the answers are not precise either.

In order to solve the above-mentioned problems, the trusted server approach has been proposed (see Fig. 1). In this approach the user's queries will be sent to a central trusted server. This server will perform spatial and temporal cloaking and send K-anonymized queries to the main server. So, the final queries are anonymized, and the intruder could not identify the user. More over since the trusted server knows the different queries from different users, it could add optimum spatial and temporal cloaking to the queries, thus the quality loss will be minimized.

On the other hand, the trusted server could become the security bottleneck by itself since all the queries are sent to it; in case of unauthorized access, the user's information could be revealed.

Using the P2P anonymization technique could prevent the risk of having a central bottleneck and in case of having a good implementation of it, the nodes could cooperate with each other to perform anonymization, and also the user's sensitive information will be preserved [5].

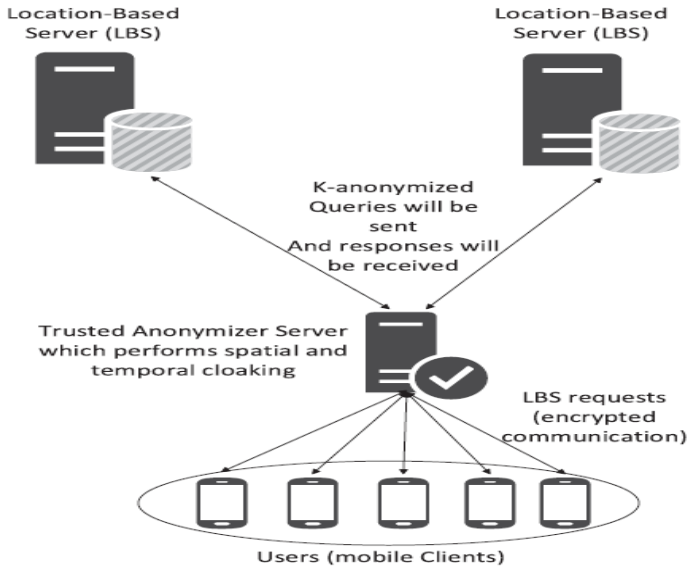

Figure 1: Central Trusted Server (reproduced from [5])

We implement a variety of anonymization techniques on user queries, including peer-to-peer anonymization techniques, such that no one can have identifiable information about specific users, which is important for privacy. We also need to implement a secure connection between the users such that no one could access the data, before anonymization. We could have two types of anonymization in this project: temporal cloaking and spatial cloaking. Students can choose either type of cloaking, or dynamic selection between these two types of cloaking, to secure user's data and privacy while minimizing quality of service loss, which is the consequence of applying anonymization.

3. CASE STUDY

IN this section, we outline an example hands-on lab for our students. **Goals** of this lab is to make sure that students

- Familiar with the concept of anonymity
- Learn how location cloaking can help preserving users' privacy
- Study about concepts of K-anonymity
- Implement and work with anonymizer

Tools and environment that we are using for this lab: Ubuntu, Wireshark and Amazon AWS.
Scenario of this hands-on lab: During an outbreak disease period (such as zika virus), this hands-on lab exercise gives you training about applying techniques that preserve anonymity, and enables you to explore the tradeoffs between preserving anonymity and privacy on one hand versus enabling the applications functionality on the other hand. In this system, users will send their information, such as hometown and current location, to the server using the application. In response, the application tells them if there is a zone with a high risk of a disease nearby. The server analyzes the information of users in a zone and assigns a risk factor to the zone based on information on users located there, such as their home location, places they have visited, and their health information. The server then announces this risk factor to all nearby users, so that they may avoid that location if they consider the risk is too high.

The challenge here is the resolution of the announced zones. A higher resolution results in high quality of service. Higher quality of service means that the application has smaller tagged zones; if a small zone is marked as risky, one could avoid it easily since it is small. However, this might violate user's privacy or anonymity. For example, if a high-risk zone contains just a few users, others observing that location can guess that people there are probably sick (Figure 2). The intruder can access user data either physically for whom are owning the server or working there or through online intrusion such as man in the middle attack or hacking the server. The man in the middle attack could happen anywhere on the way between the user and the server and since all the data is stored on the server, accessing the server means accessing the user information. This privacy concern is not desirable for users. Therefore, there is a clear trade-off between the quality of the service and privacy of the users.
In order to preserve privacy, the most common methods are: **Fake queries**: send fake queries along with the real one such that detecting the real query become hard.

Pseudonyms: Instead of using user name, use a Pseudonym as an ID for him and change it over time to make it hard to detect the real identity of the user.
Spatial cloaking: this method can be used by considering a zone around user's query and change the location of the query randomly to somewhere in that zone. This method prevents detecting the user based on his query location. For example is the query location is (X,Y) after spatial cloaking it will be (X+alpha1, Y+alpha2), where alpha1, alpha2 are two random variables in the range of (-R,R). The R is the radius of the zone that we considered as the cloaking zone around the query.
Temporal cloaking: In order to apply temporal cloaking one can consider a specific time period and change the query time randomly within that time period so the exact query time is not detectable. For example, the query will be delayed for t seconds where t is between 0 to 100 seconds.

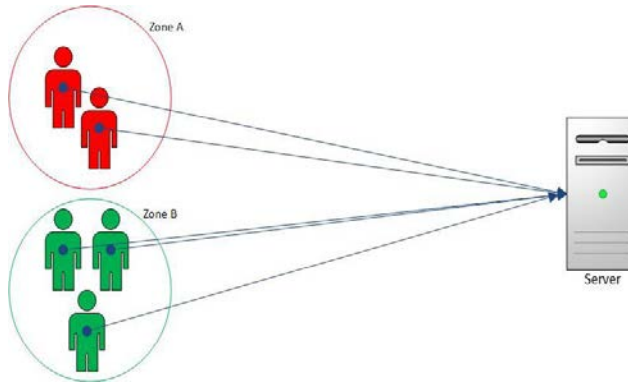

Figure 2. Defining the risk of each zone without applying anonymity could cause privacy concerns for the users

. One simple method is to use a location anonymizer. Location anonymizer is a web server which performs K-anonymity technique on the user's data. It performs the K-anonymity by applying spatial or temporal cloaking to the queries to make them indistinguishable. Therefore, the location anonymizer changes the location of the user queries such that at least K of them are indistinguishable. K-anonymity ensures that each zone has at least K users. if K is sufficiently high, then an observer cannot relate the risk of the disease to any of the users in that zone. Using a trusted anonymizer, the resolution of the zones could be changed dynamically such that each zone has at least K different users. So, the intruder cannot correlate the risk to any specific person and the privacy of the users will be preserved. At the same time the announced zone by the server will become bigger, which decreases the quality of the

service. For example, instead of announcing a house as a high-risk zone, the server may announce a part of a street as a medium risk zone.

One way of achieving K-anonymity is using a central trusted anonymizer. For this purpose, users send their query to a trusted node in the network called anonymizer (Figure 3). The anonymizer applies spatial cloaking on the queries to make them indistinguishable. For this goal, the anonymizer gets the user information and the predefined value of the K. Based on the K, we need to have at least K indistinguishable queries at each zone. So, if there are few users in a specific zone, the trusted anonymizer combines this zone with one of the adjacent zones to have at least K users in the newly created zone (Zone C in Figure 3). Then it changes the location of the users queries in this new zone to make them indistinguishable (Figure 3). After performing spatial cloaking, the location of queries is changed, so that they do not reveal the exact location of the original sender of the queries. Anonymizer then sends the queries to the server.

While the server cannot realize the exact location or identity of the original sender of a query, it can process the queries and send the responses back to the anonymizer. Finally, the anonymizer sends the responses to the users since it knows the owner of each query. Using this method, there is no direct interactions between user and the server. In this scenario, we assume that the trusted anonymizer server is safe and reliable and will not reveal those user identities or sensitive information. However, the server can store the users queries and analyze them to detect their identities and their sensitive information. Especially in the new age that the servers have access to the huge storage and computing power and novel machine learning and data mining methods.

As an example, assume we have zone A with 2 infected patients in it and zone B with 3 healthy persons in it and K was defined as 5. The anonymizer will anonymize the queries and sends the queries as zone C (which includes both zones A, B), such that in the new zone we have two infected and three healthy people. At the same time the location resolution has not been changed significantly and instead of either zone the server considers a new bigger zone which includes two zones. Using this approach, the privacy of the users will be preserved while the server is still able to send a response with an acceptable resolution.

In our proposed plan, students will connect to the cloud where the central trusted anonymizer, several user nodes, and the server are implemented. They will use Wireshark software to check the packets and understand more about the anonymizer and its benefits.

By checking the packet contents before and after anonymization, they will understand this process better. In some cases, there might be zones with a single person or a few people from one country. In these cases, having the risk factor of these zones, conclusions can be drawn about that single person, or the country of people in a zone. This can be a violation of user privacy. Students can try finding such scenarios to understand the importance of privacy preserving techniques that we are teaching in these labs.

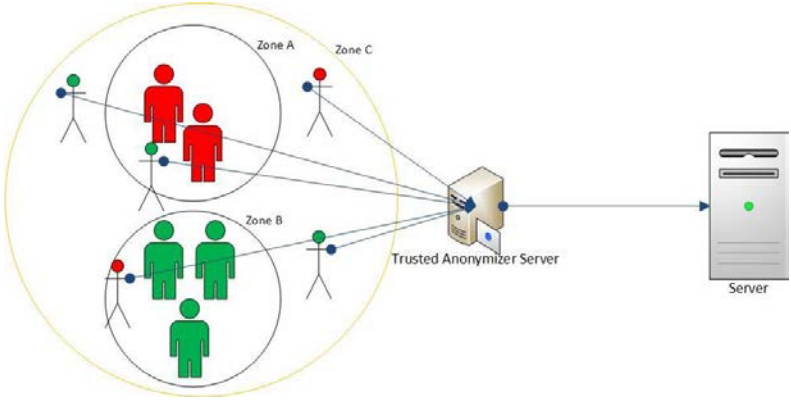

Figure 3. The anonymizer performs spatial cloaking to the user queries and send the new locations to the server. In this case the privacy of the user will be preserved.

4. CONCLUSIONS

In this paper, we were able to implement several hands-on labs based on K-anonymity and other preserving privacy methods. Feedbacks from our students are positive and promising. In the future, we will continue to improve and expand more hands-on labs to include current trending topics and popular security tools based on applying anonymity framework. In addition, we will continuously retrieve student feedback to make activities better learning tools and more student-friendly.

REFERENCES

[1] Free, Caroline, et al. "The effectiveness of mobile-health technology-based health behaviour change or disease management interventions for health care consumers: a systematic review." PLoS medicine 10.1 (2013): e1001362.

[2] Fox, Susannah, and Maeve Duggan. Mobile health 2010. Washington, DC: Pew Internet & American Life Project, 2010.

[3] Bengtsson, Linus, et al. "Improved response to disasters and outbreaks by tracking population movements with mobile phone network data: a post-earthquake geospatial study in Haiti." PLoS medicine 8.8 (2011): e1001083.

[4] Lu, Rongxing, Xiaodong Lin, and Xuemin Shen. "SPOC: A secure and privacy-preserving opportunistic computing framework for mobile-healthcare emergency." IEEE Transactions on Parallel and Distributed Systems 24.3 (2013): 614-624.

[5] Ghaffari, Meysam, et al. "P4QS: A Peer-to-Peer Privacy Preserving Query Service for Location-Based Mobile Applications." IEEE Transactions on Vehicular Technology 66.10 (2017): 9458-9469.

[6] Li, Lei, et al. "Developing Hands-on Labware for Emerging Database Security." Proceedings of the 17th Annual Conference on Information Technology Education. ACM, 2016.

[7] Peng, T., Liu, Q., Wang, G., Xiang, Y., & Chen, S. (2019). Multidimensional privacy preservation in location-based services. Future Generation Computer Systems, 93, 312-326.