Privacy-Aware Remote Mobile Health and Fitness Monitoring: Extending the Functionality of The Berkeley Telemonitoring Framework

Kaidi Du
Ruzena Bajcsy, Ed.
Ali Javey, Ed.
Daniel Aranki, Ed.

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2017-36
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2017/EECS-2017-36.html

May 8, 2017

Copyright © 2017, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Acknowledgement

Professor Ruzena Bajcsy, University of California, Berkeley
Professor Ali Javey, University of California, Berkeley
Daniel Aranki, University of California, Berkeley

University of California, Berkeley College of Engineering

MASTER OF ENGINEERING - SPRING 2017

Department: EECS

Concentration: Physical Electronics and Integrated Circuits

TITLE OF PAPER: Privacy-Aware Remote Mobile Health and Fitness Monitoring

STUDENT FULL NAME: Kaidi Du

This **Masters Project Paper** fulfills the Master of Engineering degree requirement.

Approved by:

1. Capstone Project Advisor:

Signature: Ruzena Bajcsy Date 5/11/2017

Print Name/Department: RUZENA BAJCSY/EECS

2. Faculty Committee Member #2:

Signature: Ali Javey Date 5/1/2017

Print Name/Department: ALI JAVEY/EECS

BERKELEY TELE-MONITORING

Privacy-Aware Remote Mobile Health and Fitness Monitoring: Extending the Functionality of The Berkeley Telemonitoring Framework

Master of Engineering Capstone Report

Kaidi Du

with Caitlin Gruis and Yu Xiao

Friday, May 1st, 2017

https://telemonitoring.berkeley.edu/

Acknowledgements

Professor Ruzena Bajcsy, University of California, Berkeley

Professor Ali Javey, University of California, Berkeley

Daniel Aranki, University of California, Berkeley

Table of Contents

Abstract

Unregulated sensitive data which are “not legally regulated but still considered sensitive due to proprietary, ethical, or privacy considerations,” can infer regulated sensitive data like medical history “protected under federal or state regulations” [1]. For example, an individual’s unregulated respiration rate may deduce if this individual has lung diseases, considered as regulated sensitive data. To protect sensitive data, it is therefore, necessary to protect both regulated and unregulated sensitive data. We can restrict access to all sensitive data, but what can we do if we would like to remotely transmit our medical history to doctors to allow analysis? How can we know that the privacy of our data is protected during the transmission? This paper introduces an implement of a method using Java to sanitize data which reveals as little as possible sensitive data to an unauthorized party so that the risk of privacy disclosure can be reduced.

Chapter 1 Technical Contributions

1.1. Motivation for the Berkeley Telemonitoring Project

The Berkeley telemonitoring project aims to provide a framework for developing telemonitoring apps for remote diagnosis and training purposes on Android smartphones[1]. Therefore, securing privacy plays an important role in this telemonitoring project. Our previous studies have successfully designed an telemonitory system for cardiologists to remotely monitor patients' physical conditions. In this project, our team would like to advance and generalize this telemonitoring framework on the Android platform so that it would be easily used on smartphones in a broader health-related area.

1.2. Division of Labor

The project is divided into three parts: building up the server, improving Bluetooth Low-Energy (BLE) capabilities, and sanitizing/desanitizing data before/after the storage and transmission (Figure 1). By building up the server, doctors can collect data from patients and deliver the data to the cloud using Bluetooth and send a notification to both the doctor and the patient if the data are abnormal. My contribution is using sanitization techniques to ensure privacy during the data transmissions and storages.

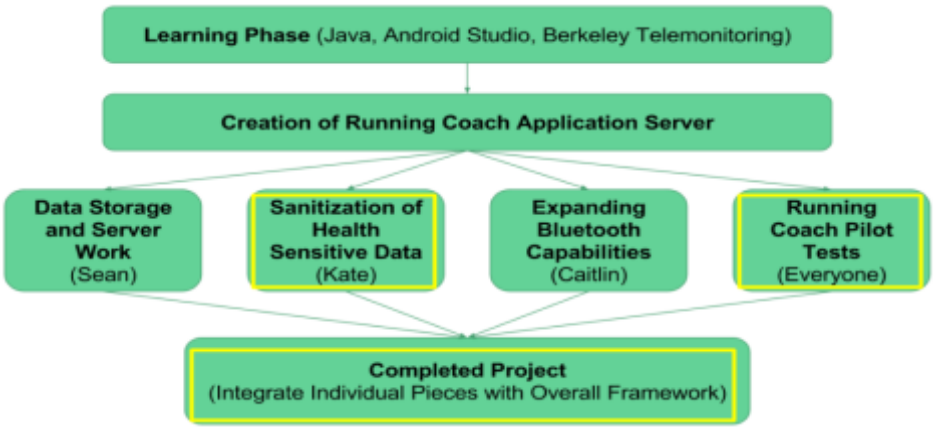

Figure 1. Division of labor in the Berkeley telemonitoring project (my parts are highlighted by boxes).

[1] Android smartphones: https://developer.android.com

1.3. Introduction

With the wide usage of electronic health records in medical institutions, the needs of protecting privacy in the telehealth area continue to grow. Sensitive data "must be controlled from creation to destruction, and will be granted only to those persons who require such access to perform their jobs, or to those individuals permitted by law" [1]. The sensitive data are classified into regulated sensitive data, which are "protected under federal or state regulations," like medical history, and unregulated sensitive data, which are "not legally regulated but still considered sensitive due to proprietary, ethical, or privacy considerations," like respiration rate [2]. However, the boundary of regulated sensitive data need to be expanded in this project because adversaries may infer regulated sensitive data from unregulated sensitive data. For example, an individual's respiration rate is usually treated as unregulated sensitive data. However, one's collected respiration rate may deduce if this individual is a smoker or has lung diseases, which will implicitly be considered as regulated sensitive data. Therefore, it is necessary to protect both regulated and unregulated sensitive data during the storage and transmission (Figure 2).

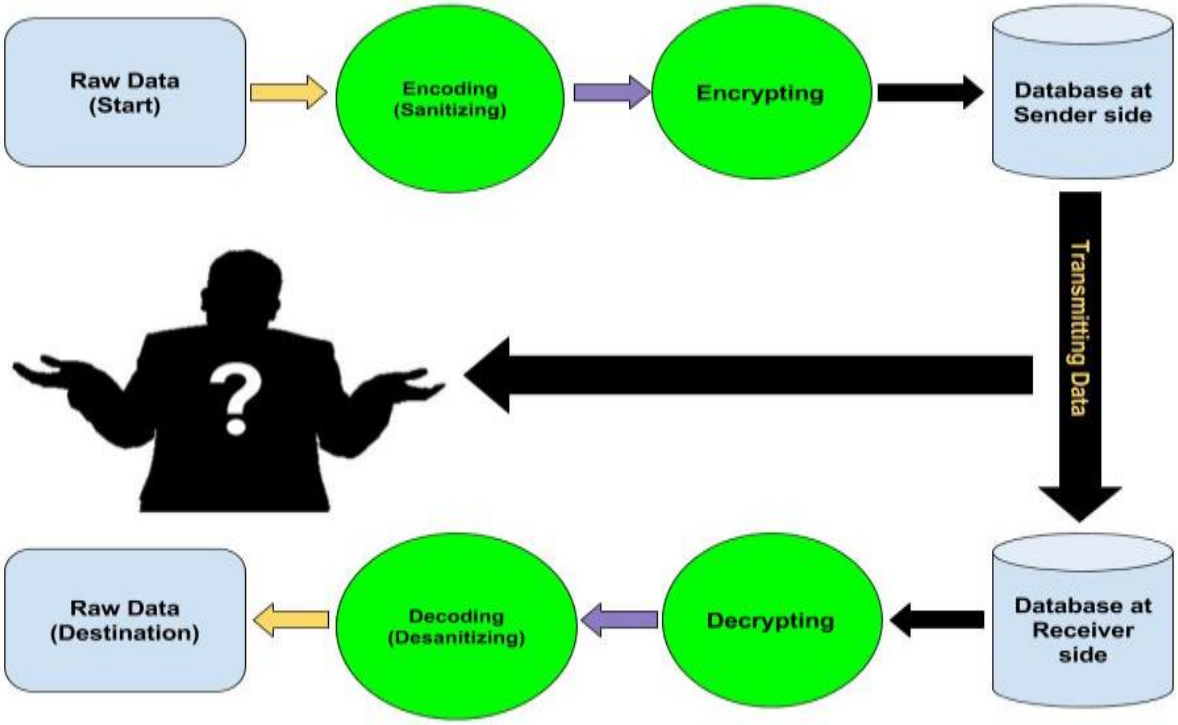

Figure 2. The sanitization/desanitization processes during transmission. Encode all the raw data and encrypt regulated sensitive data before the transmission; decrypt the regulated sensitive data and decode all the data to recover the raw data. If an unauthorized party hacks the data during the transmission, he/she will not get the real information.

Unfortunately, there is no guarantee that transmitted sensitive data will not be hacked by an untrustworthy hacker. But it is achievable to hide regulated sensitive data and make unregulated sensitive data incomprehensive to adversaries. We plan to use encoding methods to sanitize unregulated sensitive data. In other words, sanitization is a process to hide information so that the data from different classes would be indistinguishable. We will use the privacy mapping function (PMF), described by Daniel Aranki, to encode the transmitted data. This paper will focus on describing the implementations of encoding/sanitizing techniques in this project, specifically on: (a) previous related work; (b) implementation of privacy mechanism; (c) potential challenges; (d) future works.

1.4. Background Information

To keep sensitive data confidential, we should secure data in both databases and transmission processes. The owners sanitize their data in both databases and transmission processes. However, the majority of research in privacy in the health area focuses on preventing publishing medical data and is therefore database-centric [3, pp. 2]. Currently, two models of privacy-reserving methods play an important role in the medical-related privacy literature: k-anonymity [4, pp. 560] and differential privacy [5, pp. 7]. K-anonymity is a table in which the total number of records per quasi-identifiers is at least k [4, pp. 560]. This method can prevent privacy data publishing because the probability of linking a record to an individual is at most 1/k [3, pp. 2]. Similarly, the differential privacy method also "requires that the output of a statistical query should not be too sensitive to any single record in the database" [3, pp. 2]. However, these two data sanitization methods cannot prevent adversaries inferring regulated sensitive data based on a piece of unregulated sensitive data during the transmission, because it was designed to preserve private statistical databases. Our privacy mechanism would make each element from different categories in the original domain be uniformly distributed in different categories or the same category in the destination domain. Therefore, the accuracy of inference would dramatically decrease because the relationship between elements in the original domain was broken. If we can apply the same sanitization method on the transmitted data in our tele-monitoring project, users' private information would be unidentifiable and preserved during the transmission.

1.5. Data Sanitization in Transmission and Storage

Data sanitization aims to prevent adversaries from statistically inferring private information, like a health condition or a disease, based on disclosed individuals' messages during communication. We can mask the original private data in a way that they are highly indistinguishable to unauthorized parties. For the sake of brevity, one way to achieve this goal is using privacy mapping function (PMF) to sanitize/encode data without changing the data size.

This PMF encoding technique was introduced by Daniel Aranki and Prof. Ruzena Bajcsy at UC Berkeley [3].

1.5.1. Privacy Mapping Functions—Sanitization and Desanitization Functions

Generally, for unregulated sensitive data, sanitization and desanitization only include the process of encoding and decoding data using PMF functions: for every private diagnosis, we use a different sanitization function that has a left inverse, such as an affine function $f(x) = \alpha * x + \beta$.

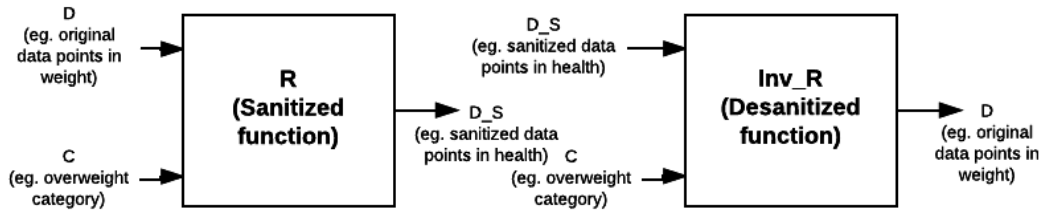

Figure 3. Both R (Sanitization function) and Inv_R (Desanitization function) are PMFs. D represents the original data points as the input data of the R (Sanitized function) and output data of the Inv_R (Desanitized function). D_S represents the sanitized data points as the input data of the Inv_R (Desanitized function) and output data of the R (Sanitized function). C represents the input category that we wish to keep private.

The idea of the PMF function is shown in Figure 3. Suppose we would like to sanitize the information about weight to hide the weight status/categories including underweight, normal or healthy weight, overweight, and obese[2]. To limit the ability to gain further information based on weights, we make different weight categories indistinguishable to each other, such as mapping different individuals weight information from different classes into the same class (Figure 4).

[2] Centers of Disease Control and Prevention: https://www.cdc.gov/healthyweight/assessing/bmi/adult_bmi/index.html

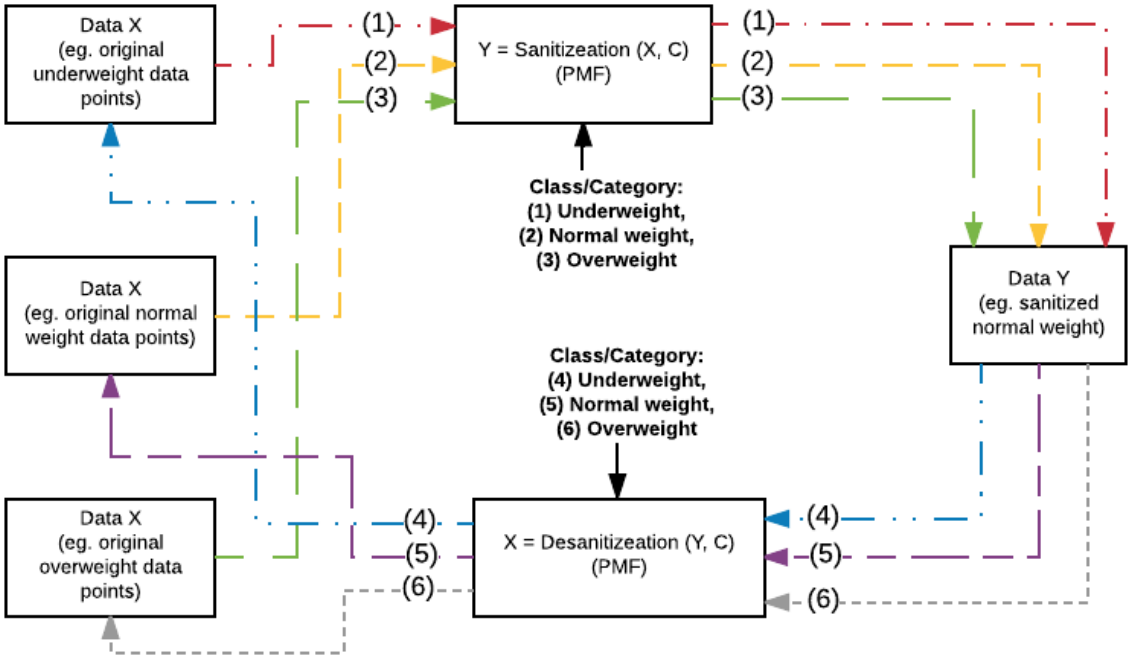

Figure 4. Lines marked by the same number, color, and line style represent a complete process of sanitization/desanitization. (1), (2), (3) represent the process of using privacy mapping function (PMF) to map/sanitize original data points (Data X) from different categories(C) into the sanitized data points (Data Y) in the same category (normal weight). (4), (5), (6) represent the process of using privacy mapping function (PMF) to recover/desanitize data points (Data Y) from the same categories(C) to the data points (Data Y) in their original categories.

However, we also would like to desanitize them to let analysts/doctors access accurate weight status information. To realize the desanitization process, we need to make some assumptions. Suppose both the patient/sender "John" and the doctor/receiver "Dr. Emily" have already known the class of weight status "overweight" and his weight related disease "hypertension". When John finished collecting his weight data points using his smart phone, he would be asked to input the weight category so that the data points (D = data points in weight) and the known category (C = overweight) would be automatically passed into the sanitizing function (R) to output sanitized data points (D_S = data points in health) in Figure 3. These sanitized data points would be encrypted and be temperately stored in the smartphone before the transmission. When Internet

connections were available, these sanitized encrypted data would be sent out to Dr. Emily and stored in her database. If anyone from unknown third party hacked these sanitized encrypted data during the transmission, even if the adversary knew the password and decrypted data, he would not be completely sure about John's real weight information and health condition without knowing his weight category. After the sanitized data were transmitted, they would be stored in the database at the receiver side. Consequently, the privacy would be preserved during data transmissions and retention.

To make raw weight data available to Dr. Emily, we need to decrypt and desanitize these transmitted data points from the database at the receiver side. Because Dr. Emily had known John's weight category (overweight), she would be able to desanitize his weight information. When she looked up John's weight information, she would input both John's weight category and his sanitized weight data into the PMF desanitized function (Inv_R) in Figure 3. Afterwards, she could get John's accurate weight data points to do further analyses and treatments on his hypertension. As we can see from the example, data sanitization would not degrade the utility of the raw data for the receiver while the raw data points were secured during the data transmission and storage.

Towards a more formal but simplified representation of PMF, we can use a trivial mathematical example to explain the basic algorithm in the PMF functions. In equation (1), R (X, C) represents the sanitization function, where X is the raw weight data point and C is the category of an individual's weight that we wish to keep private. The output Y represents the sanitized data point.

$$R(X,C) = Y = \begin{cases} \frac{6}{5}X, & C = \ underweight \\ X, & C = normal\ weight \\ \frac{5}{6}X, & C = \ overweight \end{cases} \quad (1)$$

In equation (2), $R^{-1}(Y, C)$ represents the desanitization function, which is the inverse function of R (X, C), where Y is the sanitized weight data point and C is the category of an individual's weight. The output X represents the raw weight data point.

$$R^{-1}(Y, C) = X = \begin{cases} \frac{5}{6}Y, & C = \ underweight \\ Y, & C = normal\ weigh \\ \frac{6}{5}Y, & C = \ overweight \end{cases} \quad (2)$$

If two men have same height 5'6", the weight range will be

$$a\ man's\ weight\ with\ height\ 5'6" = \begin{cases} \text{below 154 lb.}, & \text{underweight} \\ \text{155 lb} - \text{185 lb.}, & \text{normal weight} \\ \text{over 185 lb.}, & \text{overweight} \end{cases} \quad (3)$$

Assume this person's weight is about 150 lb. and is in the underweight category. If we pass this information into the sanitization function R, the result should be 180 lb. which is mapped into the normal weight class. Similarly, if another person's weight is 190 lb. and is in the overweight category, the result will be 158 lb. which is mapped into the normal weight class as well. Therefore, if someone without knowing the category/class of the weight hacked this sanitized information, the real weight information would not be exposed. Similarly, we can pass 180 lb. with the underweight category and 158 lb. with the overweight category into R^{-1} to desanitize these data points, and then we will respectively get 150 lb. and 190 lb. Consequently, we can use PMF to secure privacy and keep the utility of data. The PMF automatically generated by a computational software MATLAB. The MATLAB implements as a toolbox. Although the coefficients and constants in the real PMF functions may be different with the example, the goal is the same which is to make data from different classes highly indistinguishable.

1.5.2. Implementations of Privacy Mechanisms

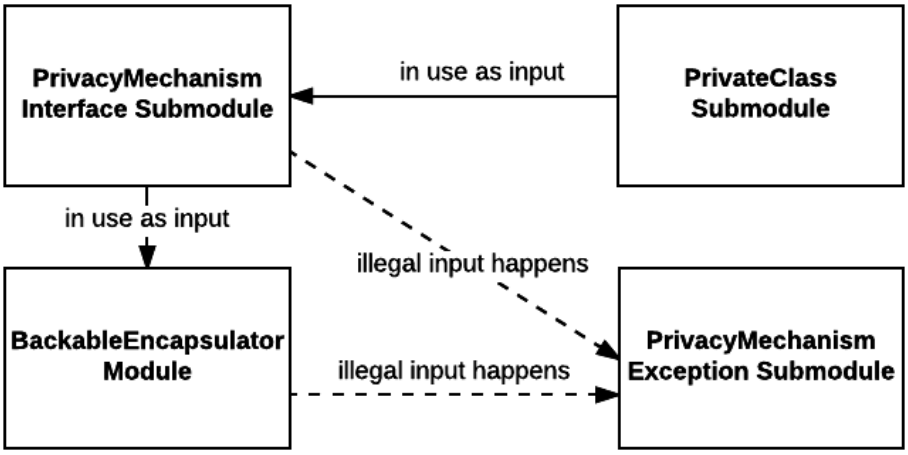

Figure 5. Hierarchical structure of the privacy mechanisms. The solid arrows represent that the module at the tail of the arrow is the input of the module at the head of the arrow. The BackableEncapsulator module represents the cabinet which includes multiple methods to store, recall, and recover data points. The PrivacyMechanismInterface submodule represents the class which includes sanitization/desanitization methods to manipulate data points. This submodule would be passed into the BackabeleEncapsulator as an input object. The PrivateClass submodule represents the class which is the type of the input category. This submodule would be passed into the PrivacyMechanismInterface submodule. The dashed arrows if the input categories of the data points are not in this type, i.e. an illegal input happens, the sanitization/desanitization will be thrown into PrivacyMechanisim Exception submodule.

The privacy mechanisms are incorporated in the core of the framework. The hierarchical structure of the privacy mechanisms (PMFs) is shown in Figure 5. We implemented the privacy mechanisms (PMFs) in the *BackableEncapsulator* module which would "allow storing and recovering this encapsulator privately and securely" [23]. The *BackableEncapsulator* module is the core for both storage and transmission data jobs. With PMFs, the original mechanism in the *BackableEncapsulator* module should still work. In other words, the PMFs should be independent of the implementation of the data structures, and the new version of the *BackableEncapsulator* module should be compatible with both new/sanitized and old/unsanitized versions of data points [3, 21]. It is similar to the situation that a user can use

Microsoft Word 2016 to edit the document created by Word 2010 without any compatibility problems. To realize this function, we apply the overloading mechanism in Java to our *BackableEncapsulator* module. In this case, the *BackableEncapsulator* module includes the old library and the new library (Figure 6).

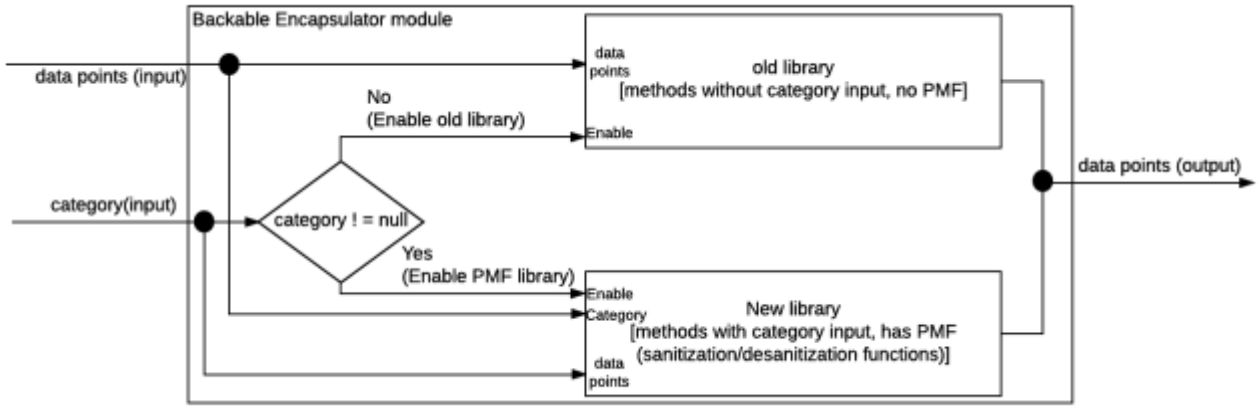

Figure 6. The Backable Encapsulator module includes both the new library and the old library. If a user inputs the category of data points for the sanitization/desanitization, the methods in the new library will be called, and otherwise, the methods in the old library will manipulate the data points.

There are multiple methods with the same signatures but different types of inputs in these two libraries. Table 1 lists details about these methods. If a user inputs the category, the method with category input will be called in the new library. Otherwise, the input data points will be passed into the old library. In other words, this PMF can manipulate both sanitized data points in the new version of our software and unsanitized data points from the old version of our software. Whatever the version of software a user uses, the data could be passed into the *BackableEncapsulator* module without any problems. For example, in the last version of our software, it did not have the sanitization method, and therefore, all data points are stored without the sanitization process. If a doctor would like to use the new version of the software to retrieve a patient's data from the old version of the software, the stored unsanitized data will still be passed into the PMF without the category input but the result will be the same with the input data. Hence,

after updating the old version software by adding the PMF there, whatever worked in the old version should work in the new version of the software as well.

	Methods
New Library (with PMFs)	public BackableEncapsulator (BackableEncapsulatorIdentifierInterface dataID, PrivacyMechanismInterface pdi)
	public long addData (T datapoint, PrivateClass c)
	public T getData (long dataSerialNumber, PrivateClass c)
	public DataSerialPair<T> popFirst (PrivateClass c)
Old Library (without PMFs)	public BackableEncapsulator (BackableEncapsulatorIdentifierInterface dataID)
	public long addData (T datapoint)
	public T getData (long dataSerialNumber)
	public DataSerialPair<T> popFirst ()

Table 1. Methods with the same signatures in both the new library and the old library. The differences between methods in the new library and the old library are the inputs. PrivateClass c input represents the category of the data points which we would like to hide; PrivacyMechanismInterface pdi input represents the object which contains sanitization and desanitization methods.

To sanitize/desanitize data points in the *BackableEncapsulator* module, we recalled the sanitization/desanitization method from *PrivacyMechanismInterface* shown in Figure 5. In the sanitization/desanitization method, the change of the category of data points was determined by the mapping algorithms in the sanitization/desanitization method. For instance, if we called the method *addData (T datapoint)*, the raw data points would be added to the container without the process of the sanitization. Conversely, if we called the method *addData (T datapoint, PrivateClass c)*, we would input the raw data points as the first input (*T datapoint*) and the category of the data points as the second input (*PrivateClass c*). Before the data points were added to the database, they would be passed into the *PrivacyMechanismInterface* and be sanitized based on the input category *PrivateClass c*. The output of the *PrivacyMechanismInterface* would be added to the database. Therefore, the privacy mechanism is realized in the *BackableEncapsulator* module.

Furthermore, we added *PrivacyMechanismException* into this privacy mechanism to handle the *Exception* and keep the sanitization process from terminating without explicitly showing the reason (Figure 5). In other words, if a user inputs a category which does not exist in the predefined category list in the PMF function, the function will return a message that says “Error, the input category does not exist” and allow the user to retry. We put this function into practice by creating

a new Exceptions Class “PrivacyMechanismException” in Java because it would handle illegal inputs without terminating the program [7]. We define the “PrivacyMechanismException” class by inheriting “IllegalArgumentException” class in Java. After we extended the PMF using this exception function, the program would increase the software fault tolerance.

Regulated sensitive data need further protection. After we encoded all sensitive data, encryption is another step for securing these regulated sensitive data so that it will be incomprehensible to adversaries. In this project, we will use a password encrypting technique called Transport Layer Security (TLS) or Secure Sockets Layer (SSL), which is a protocol that provides privacy and data integrity between the server (Java) and the clients (Android) and ensures their online communication remain confidential [22]. The details are shown in Figure 6. Specifically, we will use the password method to encrypt our data. The encoded data will be manipulated in the following four steps: (a) The client asks the smartphone to make a secure connection to the server. (b) The server obtains the IP address of the client and requests a secure connection to the client. (c) To initiate this secure connection, the client requests the server to send a copy of its SSL certificate to the client to identify itself. (d) If the client confirms the server, it would create a symmetric session key to encrypt the data being transmitted and sent it to the server [9, pp. 5]. When the data are encrypted, it will be transmitted to the receiver.

After the transmitted data arrive to the intended receiver, the receiver must input the private password and public key to decrypted data to get the encoded data again at the receiver side. The receiver can decode it using the method of the inverse PMF to get the original data. As we can see in this case, the encrypting approach is a complement to encoding data instead of an alternative way in protecting sensitive data during the transmission. If the encryption is compromised by an adversary, the encoded data can keep the unregulated sensitive data preserved. Therefore, the combination of encoding and encryption can ensure privacy.

There are two constraints in data sanitization for both regulated and unregulated sensitive data. To successfully implement encoding/PMF function, we assume both senders/patients and

receivers/doctors know the category of data points. However, if the receiver/doctor forgot to input the category into the PMF function to recall data, the program would treat the sanitized data point as raw data point and output the wrong results without decoding them. Equivalently, if a user at the receiver side input a wrong category into the PMF function to retrieve data, he/she would also get the biased results which would degrade the accuracy of data analysis. Therefore, it requires users to know the correct data category to get the accurate data points.

Another constraint in this privacy mechanism is that we assume that there is asymmetry in the knowledge between the intended recipient and the unauthorized adversaries. This assumption stated that the intended recipient had more certain knowledge than the adversaries about the transmitted data. However, if the adversaries had the almost same certain knowledge with the intended recipient, this privacy mechanism would be compromised. For example, if the adversary once knew the sender who sent the sanitized weight information to the doctor, this adversary would have known the sender's weight status/category so that the expected function of the PMFs would be compromised. Therefore, this is another constraint which we would like to overcome in the future.

1.6. Future Work

In the future, we will add privacy-aware data analysis tools to the server side to justify the result of data sanitization. We will use the MySQL database, which was designed by the whole capstone team, to store our sanitized data and compute the probability to get accurate raw data from the sanitized one. To be specific, we will use machine learning algorithm to train our prediction model in three classifiers with sanitized data. Afterwards, we will use this model to test our sanitization results and calculate the quotient between the number of correctly predicted data and the total number of predicted data. If the total accuracy can be closed to the trivial classifier which always predicts the largest class (deterministic), our data sanitization has succeeded to protect the subjects' privacy.

1.7. Technical Contribution Conclusions

In this paper, we presented a data sanitization technique to secure privacy in communications and storages. It is clear that no one wants adversaries to acquire their sensitive data. To improve privacy in this Berkeley tele-monitoring project, we used encoding techniques for sanitizing unregulated sensitive data. In addition, we used both encoding and encrypting techniques for sanitizing regulated sensitive data. These techniques can not only hide regulated sensitive data to prevent privacy exposure, but also protect unregulated sensitive data to avoid the potential risk of inferring regulated sensitive data from it by untrusted people. Although there may be some constraints in this sanitization technique, such as pre-assuming category limitation and asymmetrical knowledge between the recipient and the unauthorized parties, we will add more robust algorithms into the PMF and use machine learning algorithms to do further evaluations in the future.

Chapter 2 Engineering Leadership

2.1. Engineering Leadership Introduction

We now turn from technical contributions to a new topic of engineering leadership, with the goal of viewing our project from a business perspective. For this portion of the report, we consider bringing the Running Coach application to the market. This application is an instantiation of our open source health telemonitoring framework. It focuses on creating coaching plan by monitoring user's health condition, such as heart rates and blood pressure. With this application comes several questions that need to be answered regarding where our product fits into industry, how we would market it, and social implications.

2.2. Industry Analysis: Marathon Running Coaching

Today's world is one that is increasingly health conscious. Over 50% of Americans exercise on a regular basis, and this number is on the rise [10]. In addition to this, over 3.3 million fitness bands and exercise trackers were sold between March 2013 and April 2014 in the U.S. [11]. Based on these statistics, it is clear that a change in American health patterns is on the horizon, and Berkeley Tele-monitoring believes that it can help by entering into the sports coaching industry.

The sports coaching industry is defined as "consisting of establishments that offer instruction for athletic activities to groups or individuals" [12, pp. 5]. This industry tends to target selling towards adolescents and young adults from ages 10-29. Recently, it has been doing quite well, with an average annual revenue increase of over 3.0%. This is believed to be due to an increase in sports participation [12].

There are many competitors in the sports coaching industry, and we will look at a few of them. The biggest subset consists of human personal trainers and fitness coaches at gyms and athletic facilities. However, a fast-growing portion of this industry is sports coaching software technology, and this is where our product has the most competition. These platforms are known to give teams and individuals that adopt them a competitive advantage in the form of athletic improvement.

Major companies in this industry include Coach.me, AthleticLogic, and Coach's Eye [13]. Upon reviewing these products, they appear to gather a lot of data, whether through filming an athlete doing a sport or collecting sensor data. Many of them involve recording a training session and allowing an athlete to go back and view their workout later. Where these products fall short is that they fail to provide real-time feedback to the athletes, and they use limited sensors to gather their information. Our product seeks to improve upon these downfalls by providing real-time suggestions to athletes during exercising and using a wide variety of sensors such as accelerometers, GPS, and cameras.

Overall, our final product would fit well into the sports coaching industry. Within this industry, there are several major competitors, some of which are well-established corporations. However, our team believes that we have a unique value proposition in creating a framework instead of focusing on a physical wearable device, as well as providing real-time feedback to our users for how to improve their workouts. These advantages described above play a large role into how we will choose to market our product.

2.3. Market Strategy

Our primary clients are long-distance marathon runners. In recent years, marathon events have continuously attracted a lot of professional participants and public attention. According to the Running USA Annual Marathon Report, there were 1,100 U.S. Marathon events and more than 500,000 finishers each year from 2014 to 2015 [14]. Good performance usually is due to high efforts. In order to get good results in marathon competitions, runners must have effectively practicing and coaching plans. This means that there are many potential clients in the long-distance runner coaching market. A Marathon runner's performance heavily relies on their training, and therefore, a comprehensive training plan plays an important role in his/her good competition outcome. However, current personal one-on-one training programs are expensive and without many flexible schedules. For example, an in-person coach training program in COACHUP is about $100/session [14]. Although these human coaching programs can customize

the training plans, they are not able to report real-time feedback regarding performance metrics if the runner does outdoor training. However, our smartphone coaching application can quickly generate a personalized training plan for a runner for a low price. This gives our product a competitive advantage in the market.

As mentioned in the industry analysis, there are many sports coaching competitors in this market. However, we would market our tele-monitoring Android platform framework by showing how it is distinguished from the rest. The IBISWorld Industry Report 61162 showed that there were about 123,094 businesses in the sports coaching field [12, pp. 4]. Many of these businesses are one-on-one sports training camps and schools that offer instruction in athletic activities to groups or individuals [12, pp. 2]. Additionally, the current health and fitness apps are designed for general sports or fitness purpose instead of specializing in long-distance running training, so the measurement of performance may not be as accurate. For example, the Garmin FR620 was an advanced running GPS watch [15]. But, it could not support getting any dynamic running information [15]. Our framework design allows implementing user interfaces, data acquisition, data storage, and proper security and privacy mechanisms using APIs (application programming interfaces) through the sensors in the Android based smartphones [1]. To be specific, our design will customize the training plan by monitoring the user's heart rate based on the cadence (steps per minute) using the smartphone. This method is specifically designed for long-distance runner training and is easily implemented without compromising human coaches' schedules. It would bring more convenience to the users than traditional human coaching.

In summary, our team's marketing strategy involves analyzing the current holes in the sports coaching industry and identifying how our product can fill those needs. In considering these needs, however, we must also consider their overall social implications.

2.3. Social Context

One of the biggest social impacts we see our project having is in the world of health care. While many countries have regulations for the health care of their people, it still remains a growing

concern. Particularly in the United States, the increasing cost of health care has lasted for many years. As indicated in the report of the 2016 Milliman Medical Index, “The rate of increase is still well above growth in the consumer price index (CPI) for medical services, and far surpasses the average 2% annual increase in median household income between 2004 and 2014.” [16, pp 1]. One of the main ideas given to reduce this cost is to reduce the readmission rate for chronic health conditions [17. The high readmission rate means people have to go back to medical facilities to consult with their doctors several times, increasing the cost of their health care.

Since our project provides the possibility that doctors and patients can communicate remotely without going to medical facilities multiple times, it could significantly decrease the readmission rate, which may reduce health care costs. Our project could potentially change the health care structure over time. Instead of going to medical facilities multiple times, doctors could receive health data from their patients via our Android framework, analyze it, and give feedback if necessary. This would save patients unnecessary trips to the doctor’s office.

Outside of healthcare, our Android platform based project also follows a technological trend. According to a research paper, the Android platform market share of the smartphone was 45.4% worldwide in 2015, while secondary platform IOS only had 15.3% of the market share - three times less than Android [18]. With such a huge market share, our Android platform based project could have many more potential customers than any other platform. Also, the market share by Android could increase in the future with the new release of Google Android features later this year [19]. Furthermore, the trend of increasing use of smartphones has continued for many years. The sales growth rate of the smartphone in China for 2015 is 52% more as compared to 2014, and this rapid growth is projected to continue in following years [20]. This means that not only will the market share of Android increase, but the total number of smartphone users will go up as well.

By following the current social trends, we can conclude that there is potential for wide use of our product. This will help more patients have a better healthcare experience with a lower cost.

By taking advantage of Android's large market share of smartphones, our project could potentially change the traditional health care methods for the many people in the future.

2.3. Engineering Leadership Conclusions

In brief, our Extended Platform for Android Tele-monitoring product should perform well in the sports coaching industry due to its unique value proposition of providing real-time feedback through an Android framework. Our major buyers are long-distance runners, and marketing patterns show this group increasing every year. As a good competition outcome highly relates to a good training plan, long-distance runners will be willing to spend less money buying this smartphone based application to achieve the same high-level competition results with coaching by human trainers. Finally, in a social context, we believe our product can take advantage of the large Android market share and potentially create a positive impact on healthcare worldwide.

References

[1] “Data Classification.” (n.d.). *University of Oregon Policy Library*. [Online]. Available: https://policies.uoregon.edu/vol-4-finance-administration-infrastructure/ch-6-information-technology/data-classification. [Accessed March. 12, 2017].

[2] “Sensitive Data Classification.” (n.d.). *Safe Computing*. [Online]. Available: https://www.safecomputing.umich.edu/protect-the-u/safely-use-sensitive-data/sensitive-data-classification. [Accessed March. 12, 2017].

[3] D. Aranki, R. Bajcsy. “Private Disclosure of Information in Health Tele-monitoring”, *Cryptography and Security*, to be published. [Online]. Available: https://arxiv.org/pdf/1504.07313v1.pdf. [Accessed March. 12, 2017].

[4] L. Sweeney. “k-anonymity: A model for protecting privacy”. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002. [Accessed March. 12, 2017].

[5] C. Dwork. “Differential privacy.” *Automata, languages and programming*, pages 1–12. Spring, 2006. [Accessed March. 12, 2017].

[6] L. M. CLARK, “Sanitizing Data to Prevent Disclosing Exact Network Topology”, M.S. thesis, Univ. of California, Davis, Davis, CA, U.S., 2007. [Online]. Available: https://pdfs.semanticscholar.org/489f/1cf00a0831243f497bf5583928b6f0390db8.pdf. [Accessed March. 12, 2017].

[7] “Class IllegalArgumentException”, *Java SE Documentatio,*. 2016. [Online]. Available: https://docs.oracle.com/javase/7/docs/api/java/lang/IllegalArgumentException.html [Accessed: March. 12, 2017].

[8] “Backwards Compatibility.” *Android Developer*, 2017. [Online]. Available: https://developer.android.com/design/patterns/compatibility.html [Accessed: March. 12, 2017].

[9] A. M. White., A. R. Matthews., K. Z. Snow., F. Monrose., "Phonotactic Reconstruction of Encrypted VoIP Conversations: Hookt on Fon-iks", presented at *2011 IEEE Symposium on Security and Privacy (SP),* Oakland, CA, 2011. doi: 10.1109/SP.2011.34. [Accessed: March. 12, 2017].

[10] "Exercise Trends Market Report-US-October 2016", *Mintel*, Oct, 2016. [Online]. Available: http://reports.mintel.com/display/747876/ [Accessed Oct. 14, 2016].

[11] T. Danova, "Just 3.3 million fitness trackers were sold in the US in the past year", *Business Insider,* May, 2014. [Online]. Available: http://www.businessinsider.com/33-million-fitness-trackers-were-sold-in-the-us-in-the-past-year-2014-5 [Accessed Oct. 14, 2016].

[12] R. Masterson, "IBISworld Industry Report 61162: Sports Coaching in the US", *IBISworld*, June, 2016. [Online]. Available: http://clients1.ibisworld.com/reports/us/industry/default.aspx?entid=1542 [Accessed Oct. 14, 2016].

[13] R. Tiwari, "Sports Coaching Platform Technology Market", *PR Newswire*, Dec, 2015. [Online]. Available: http://www.prnewswire.com/news-releases/sports-coaching-platform-technology-market-worth-864m-by-2021-561558701.html [Accessed Oct. 14, 2016].

[14] "Running USA", *2015 Running USA Annual Marathon Report*, May, 2016. [Online]. Available: http://www.runningusa.org/marathon-report-2016?returnTo=main [Accessed Oct. 14, 2016].

[15] D. Rainmaker, "Garmin Forerunner 620 In-Depth Review. *DC RAINMAKER", DC RAINMAKER,* Nov. 4, 2013. [Online]. Available: http://www.dcrainmaker.com/2013/11/garmin-forerunner-review.html [Accessed Oct. 14, 2016].

[16] C. Girod, S. Hart, S. Weltz, "2016 Milliman Medical Index", *Milliman,* 2016. [Online]. Available: http://www.milliman.com/uploadedFiles/insight/Periodicals/mmi/2016-milliman-medical-index.pdf [Accessed Oct. 14, 2016].

[17] "ObamaCare Facts: Facts on the Affordable Care Act", *ObamaCare*, 2016. [Online]. Available: http://obamacarefacts.com/obamacare-facts/ [Accessed Oct. 14, 2016].

[18] I. Burguera, U. Zurutuza, S. N. Tehrani, "Crowdroid: Behavior-Based Malware Detection System for Android", *CCS'11 the ACM Conference on Computer and Communications Security,* Chicago, IL, 2011. [Online]. Available: https://pdfs.semanticscholar.org/d13d/cf524d7a96255b8c89db9db77c408190f079.pdf [Accessed: Oct. 14, 2016].

[19] A. Dhebar, "Bringing new high-technology products to market: Six perils awaiting marketers", *Business Horizons,* pages 713–722, Nov. 2016. [Online]. Available: http://doi.org/10.1016/j.bushor.2016.08.006 [Accessed Oct. 14, 2016].

[20] B. Li, H. Fu, "Research on the developing trend and strategies for mobile marketing", *2014 11th International Conference on Service Systems and Service Management (ICSSSM),* Beijing, China. doi: 10.1109/ICSSSM.2014.6874110. [Accessed: Oct. 14, 2016].

[21] D. Aranki and R. Bajcsy, "Differential disclosure of information", *EECS Department, University of California, Berkeley,* Tech. Rep. UCB/EECS-2014-47, May 2014. [Online]. Available: https://www2.eecs.berkeley.edu/Pubs/TechRpts/2014/EECS-2014-47.pdf [Accessed: Oct. 14, 2016].

[22] "What is an SSL/TLS Certificate?", *instantssl*, Oct, 2016 [Online]. [Accessed: March. 12, 2017].

[23] D. Aranki and R. Bajcsy, “A Telemonitoring Framework for Android Devices”, presented at *2016 IEEE First Conference on Connected Health: Applications, Systems and Engineering Technologies*, Washington, DC, 2016. doi: 10.1109/CHASE.2016.28. [Accessed: March. 12, 2017].