User-Device Physical Unclonable Functions (UD-PUFs) based on Mobile Device Touchscreen Pressure

Timothy M. Dee
Electrical & Computer Engineering
Iowa State University
Ames, IA, USA
deetimothy33@gmail.com

Akhilesh Tyagi
Electrical & Computer Engineering
Iowa State University
Ames, IA, USA
tyagi@iastate.edu

ABSTRACT

Described in this document is a physical unclonable function (PUF) utilizing the variability derived from the pressure with which users interact with their mobile device touchscreens. We illustrate how a sequence of these pressure values from descrete touchscreen interactions may be used to uniquely characterize a user-device pair. This characterization method may find many applications in protecting access to a mobile device from a malicious party. As a result, the effectiveness of this scheme is described in terms of how one user may be differentiated from another.

1. INTRODUCTION

Mobile devices are ubiquitus in the modern world. These devices are becoming progressively more important for many applications with security sensitive data. Securing mobile devices poses unique challenges and opportunities compared to traditional data security where it is difficult for an attacker to access the physical device on which the data is stored or from which the sensitive data may be accessed. Although there is increased probability that a device may be compromised, there is also greater number of available sensors to measure the variablility in the way users interact with mobile devices compared compared to conventional computing technology.

The reality that an attacker may be able to gain access to a physical device makes securing any data stored on or accessed by a mobile device significantly more challenging. Traditional physical unclonable functions (PUFs) which can only be used to uniquely to a given hardware device are no longer sufficient to guarentee the authenticity of a user. This modivates an extension of the traditional PUF known as a user-device physical unclonable function(UD-PUF). This UD-PUF entangles the physical characteristics of the user in combination with the device to enable a more secure authentication scheme.

A UD-PUF is a function of both the hardware of a given device and the user of that device. Such a function must change significantly given unique user-device pairs, thus we must identify a property or properties which vary among mobile device hardware and a property or properties which vary among users of a given device. The best candidates to use will be properties which present with the most variability, and properties which are most easily exposed to the android operating system.

2. RELATED WORK

3. TOUCHSCREEN PRESSURE

If our goal is be be able to distinguish a user's interaction with a given device from this same user on a different devie and from different users on any device than our description of the user will need to be a product of both the user and the device. In the android operating system there exists a pressure function which returns a value proportional to current at sides of phone for a given touch screen interaction. [1] We will henseforth refer to this as touch pressure.

The pressure function is significant, because it's value not only depends on the characteristics of the device but also on the way in which a user interacts with a given device. The effect of a given device on the touch pressure value will differ significantly due to variations implicit in the manufacturing process for the touchscreens of these devices. [**?**] Our supposition is a given user will interact with a touchscreen in such a way as to cause significant variations in the touch pressure values when compared to other users [**?**]. Given this, we have chosen touch pressure as the basis for our UD-PUF.

4. MODELING A USER-DEVICE PAIR

Interactions between useres and devices are complex. To interpret these actions in a meaningful way, in order to preform an authentication for example, it is necessary to simplify these interations. The chosen model must provide sufficient entropy such that a model generated with a given user-device pair is not consistantly reproducable by another user or on a different device. The modeling method must also be easily reproducable by the origional user on the origional device. A model having the neccessary characteristics required for this application is a Marcov Chain.

Marcov Chains are useful in predicting systems who's behavior can be modeled in descrete states. The transitions

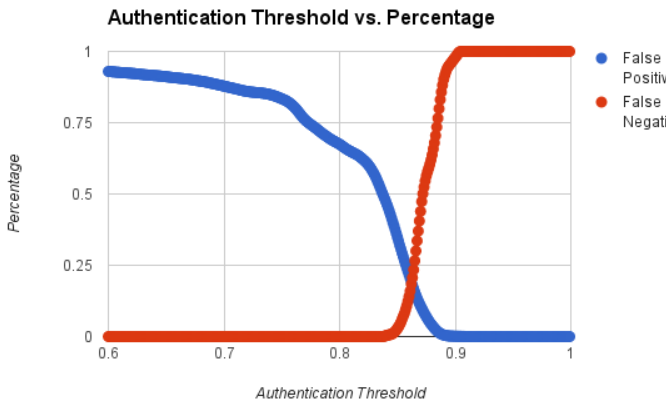

Figure 1: Describes how false positive and false negative percentages vary as the authentication threshold is adjusted.

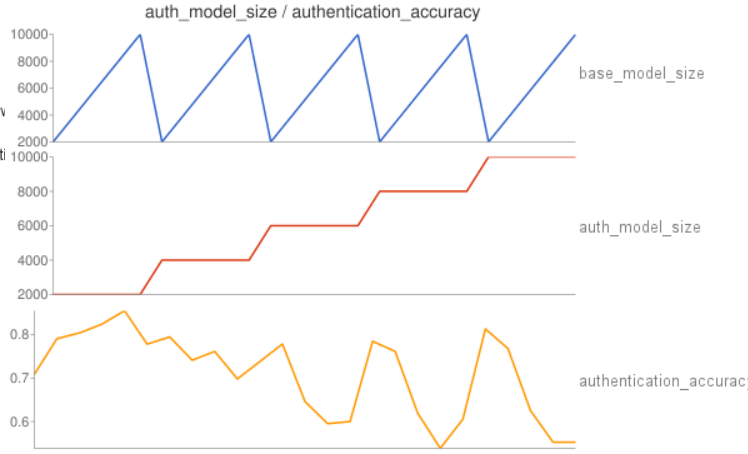

Figure 2: Authentication accuracy is a function of both the base model size and authentication model size.

between states can be identified to happen with some probability.

Historically the Marcov Chain has found applications in Upon identification of an appropriate model the next step is to discover an optimal way in which it may be applied to the current problem. An interaction between a user and device can be described as a sequence of touch pressure values. Using a Marcov Chain to describe this sequence is only reasonable if we suppose that a given touch pressure value depends on some number of preceeding touch pressure values. [**?**]

5. TOUCH PRESSURE MODELING

The goal in modeling a system with a marcov model is to classify the system in terms of its transitions between states. If such a model is to be used to purposes of uniquely identifying a given system, than the states of the model must be chosen in a way which exposes the uniqueness of the system. The states of our marcov model are defined by the range in which

6. DATA COLLECTION

Data for creating touch pressure models in this experiment was generated using a special keyboard application for the android operating system. Users would load the keyboard onto their device and continue using the device in the way they would normally. Some users were asked to play a typing game in order to help expediate the data collection process. After the users had generated atleast ten thousand touches the data was collected from the user's device.

7. DIFFERENTIATING USER-DEVICE PAIRS

In distinguishing a particular user from another different user, it is necessary to develop a method of compairason between users. In our method of compairason we take the probability associated with a touch pressure coming after a sequence of preceding touch pressures for a particular user and compute the difference between this probability and the probability of the same touch pressure coming after the same sequence of touches for a different user. The average of these probability differences is taken to be the difference between two users. Once a compairason is established a natural extension is a system of authentication. This system needs to determine when two sets of touch pressue values came from the same user-device pair. When authenticating a user, we take one minus the average difference between the model constructed from the two sets of touch pressure values. Take this value to be the authentication percentage for a given set of touch pressure values against another. To determine how well this system does at differentiating users it is useful to develop metrics which describe the system's performance under conditions which are similar to it's protential real-world applications. Figure 1 illustrates how false positive percentage and false negative percentage vary based on where the threshold for authentication is set.

Here, authentication threshold refers to the value of authentication percentage one model must have against another for the models to be considered the same; two models which are the same are supposed to have been created from touches generated by the same user-device pair. False positive percentage measures what fraction of authentcations between two sets of touch pressures which did not come from the same user-device pair, therefore these sets should not be considered the same, but did authenticate as being the same in our authentication system. False negative percentage is exactly the inverse of false positive percentage in that it describes what fraction of authentications between two sets of touch pressures which did come from the same user-device pair, but did not authenticate as being the same in our authentication system.

In Figure 1 there exists a clear intersection between false negative and false positive percentages. This intersection is significant; at this point the system nither biased toward allowing user-device pairs which should not authenticate to pass authentication nor toward disallowing user-device pairs, which should authenticate, from passing authentication. This point represents a balance allowing for

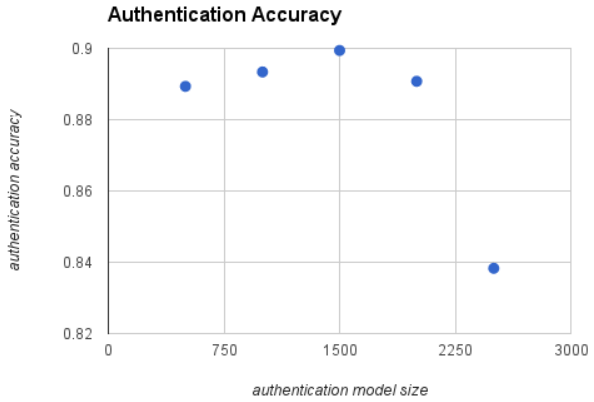

Figure 3: Depicts the result of many model compairasons done around the area of best results in Figure 2.

8. RESULTS

9. CONCLUSIONS

10. FUTURE WORK

11. REFERENCES

[1] J. Zhu, P. Wu, X. Wang, and J. Zhang. Sensec: Mobile security through passive sensing. In *Computing, Networking and Communications (ICNC), 2013 International Conference on*, pages 1128–1133. IEEE, 2013.