Authentication Feature and Model Selection using Penalty Algorithms

Rahul Murmuria
Kryptowire
rahul@kryptowire.com

Angelos Stavrou
Kryptowire
angelos@kryptowire.com

ABSTRACT

Continuous Authentication (CA) is the process of verifying the identity of the user of an electronic device repeatedly while the device is in use. Existing research in the field employs metrics such as Equal Error Rate (EER) and/or the Receiver Operating Characteristic (ROC) to evaluate the performance in the same way as 'entry-point' biometric authentication schemes. These metrics have various shortcomings with regard to CA as they fail to model the practical implications of the authentication process. We would like to discuss and get feedback on performance evaluation techniques that capture practical aspects of the authentication system including the length and frequency of times an impostor reaches different authentication levels and similarly for the genuine user. Our preliminary results show that a multi-level authentication system is not only more accurate than a binary diagnosis but it allows for high level of accuracy. We posit that further research is needed in developing such a metric for truly evaluating a CA system.

1. PROPOSED APPROACH

We use a profile generation algorithm discussed in Murmuria et al. [3] that detects events constituting significant deviations from a set of normal observations for a genuine user. The idea is to compute a measure of uniqueness or strangeness for every observation (touch gesture). We achieve that by computing the sum of the euclidian distances to the observation's k closest neighbors.

Feature Extraction The following 25 features were evaluated for each touch gesture: Average, standard deviation, range, interquartile range, and median skewness of finger diameter, pressure, finger speed, and acceleration, followed by time since previous gesture, duration of gesture, distance between end-points, arc-length of gesture, and direction between end-points.

We designed a feature selection technique that evaluates these features for performance against a dataset of 110 users using a device over a week uncontrolled. We picked a representative application - WhatsApp. In our 110 users' dataset, this app had the most data, with 40 users having over 2 hours of usage each, with at least 350 swipes and 350 taps. In order to select the best feature set, we ran a brute force search on feature subsets with a length of 1 to 5 features. This produced 83681 combinations. We processed each one of them for taps and swipes separately. We can further prune the combinations by employing smarter feature space search techniques but we left that discussion as out of scope for this paper. For training, we used 100 gestures of taps and 150 gestures for swipes to create 2 baseline models. For testing, we used a fixed size of 200 taps and 200 swipes from every user, genuine or imposter. StrOUD was applied on this set of users, comparing every user to each of the baselines. As a result, for taps and swipes, each test user produced a series of accepts and anomalies, represented as a 0/1 sequence.

Penalty Algorithm In order to calculate the final authentication scores, first we calculated penalty scores. These are calculated by assigning reward/penalty to every event in the 0/1 sequence. After assigning, cumulative sum bounded on 0 and 100 was calculated resulting in a continuous series of scores. This was done using fixed penalty and reward but these can also be proportional to other factors such as the strangeness score from the StrOUD algorithm.

Model Parameters and Mapped Scores The StrOUD algorithm requires 2 parameters: value of k for computing k closest neighbors and the confidence level with which to reject a test event in the hypothesis test. The fixed penalty algorithm requires values for penalty and reward. In addition, for every user's profile, we mapped all the scores below a threshold to 0, and all the scores above the threshold to the range of 1-100 in order to bring all users' scores along the same ranges. This threshold was found optimally for every baseline user within the range of 10-50. This resulted in 5 sets of parameters for which a grid was used to repeatedly analyze the models for all parameter combinations.

Weighted Multi-level Response After the penalty scores are mapped, we assigned weights to selected bins:

Score Bins:	0	[1,50)	[50, 60)	[60, 80)	[80, 100]
Weights:	0	0	2	5	20

Once the weights are assigned, the cumulative average is calculated. This we define as the weighted accept score (WAS). In order to get a single number to rank the parameters and feature sets with, we used the following formula for the weighted accept scores from genuine users and im-

Symposium on Usable Privacy and Security (SOUPS) 2016, June 22–24, 2016, Denver, Colorado.

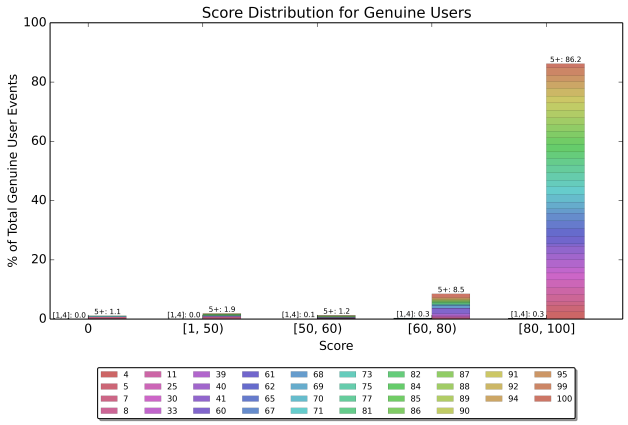

Figure 1: Score distribution for genuine users

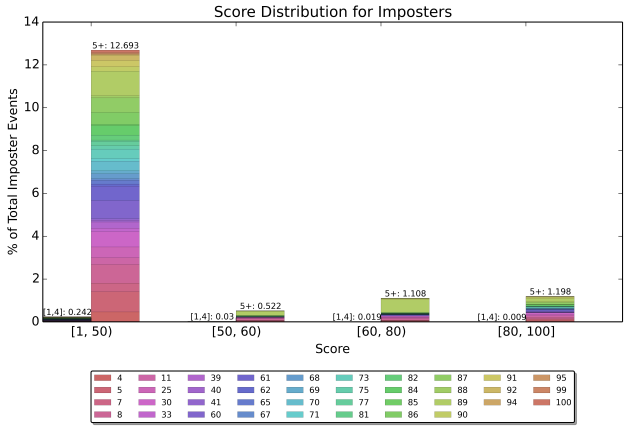

Figure 2: Score distribution for imposters

posters. The imposter score is assigned an arbitrary weight of 10 in order to give it higher importance than the genuine user score.

$$Genuine\ User\ WAS - Imposter\ WAS * 10$$

We sorted the list of parameter sets and feature sets using this score and selected the best. Further, consecutive length of each level was calculated by counting the number of consecutive values from each of the score bins.

2. PRELIMINARY RESULTS

We would like to discuss how we can achive the best feature sets and parameters for each user profile. In our studies, the best feature sets had the following features in common:

For swipes: arc-length of gesture, direction between end-points, average finger diameter during gesture, average pressure during gesture, average finger speed during gesture.

For taps: duration of gesture, direction between end-points, average finger diameter during gesture.

The best StrOUD parameters were: k value as 3 and confidence level at 75%, and the best penalty values were a reward of 0.75 and penalty of 1.5. Further every user got a different score mapping threshold that best suited the data collected for that user.

In terms of overall performance, swipes and taps by genuine user were together distributed as shown in Figure 1, whereas, swipes and taps by imposters are shown in Figure 2.

3. RELATED WORK

There is a large body of user behavior based research on CA systems for mobile devices. For the purpose of this paper, we focus on touchscreen-based systems. Frank et al. [1] evaluated each touch stroke independently and used the genuine user / imposter diagnosis to calculate False Accept Rate (FAR), False Reject Rate (FRR), and Equal Error Rate (EER). They use a notion that successive strokes when pooled together improve the confidence of the diagnosis. However, they do not implement a scoring system. Shi et al. [4] discuss a technique to fuse data from multiple sensors to create an authentication score. They do not use a traditional learning algorithm and do not need learning optimum features or parameters.

Mondal et al. [2] presented a trust model that is similar to the penalty algorithm presented in this paper. They measure the performance of the system in terms of Average Number of Genuine Actions (ANGA) and Average Number of Impostor Actions (ANIA) before detection, which captures the length of time of infiltration. However, they do not have a notion of feedback of this response to train their classifiers and they do not evaluate the level of trust that an imposter is able to gain with the system.

4. CONCLUSIONS

We have presented a novel technique to train user profiles using as feedback: (i) length of time an imposter is able to infiltrate the system, and (ii) the level of authentication the imposter is able to sustain. To that end, we proposed a metric which in addition to correct diagnosis of outliers, facilitates these additional goals.

5. ACKNOWLEDGMENTS

This paper was partly supported by DHS S&T contract D15PC00178 and DARPA contract FA8750-15-C-0056. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the DHS, DARPA, AFRL, or the US government.

6. REFERENCES

[1] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *Information Forensics and Security, IEEE Transactions on*, 8(1):136–148, 2013.

[2] S. Mondal and P. Bours. A computational approach to the continuous authentication biometric system. *Information Sciences*, 304:28–53, 2015.

[3] R. Murmuria, A. Stavrou, D. Barbará, and D. Fleck. Continuous Authentication on Mobile Devices Using Power Consumption, Touch Gestures and Physical Movement of Users. In *Research in Attacks, Intrusions, and Defenses*, pages 405–424. Springer, 2015.

[4] E. Shi, Y. Niu, M. Jakobsson, and R. Chow. Implicit authentication through learning user behavior. In *Information Security*, number 6531 in Lecture Notes in Computer Science, pages 99–113. Springer, 2011.