2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)

Continuous Transparent Mobile Device Touchscreen Soft Keyboard Biometric Authentication

Timothy Dee
Electrical & Computer Engineering
Iowa State University,
Ames, IA, USA.
timdee@iastate.edu

Ian Richardson
Electrical & Computer Engineering
Iowa State University,
Ames, IA, USA.
ian.t.rich@gmail.com

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

***Abstract*—Mobile banking, shopping, and in-app purchases utilize persistent authentication states for access to sensitive data. One-shot authentication permits access for a fixed time period. For instance, a [user name, password] based authentication allows a user access to all the shopping and payments data in the Amazon shopping app. Traditional user passwords and lock screens are easily compromised. Snooping attacks – observing an unsuspecting user entering passwords – and Smudge attacks – examining touchscreen finger oil residue – enable compromised user authentication.**

Mobile device interactions provide robust human and device identity data. Such biometrics enhance authentication. Behavioral attributes during information input constitute the password. Adversary password reproduction difficulty increases since pure observation is insufficient.

Current mobile continuous authentication schemes use, among others, touchscreen swipe interactions or keyboard input timing. Many of these methods require cumbersome training or intrusive authentication. Software keyboard interactions provide a consistent biometric data stream. We develop biometric profiles using touch pressure, location, and timing. New interactions authenticate against a profile using a distance metric. Classification achieves 100% accuracy in 3840.33 milliseconds on Nexus 7 tablets.

I. Introduction

In this paper, we characterize a soft keyboard based biometric profile – a user behavior identity. A profile is constructed from past soft keyboard interactions. It is dynamically updated using the most recent soft keyboard interactions. Validating the most recent soft keyboard interactions against this profile results in continuation or denial of device access.

Touch interactions produce electrical current flow change in a sensor grid. The Android framework reports this current as pressure. We develop a user profile from these pressure token streams over a period of time.

Physical Unclonable Functions (PUF)s further enhance biometric schemes. A PUF is a challenge response function. Providing a challenge produces a response. Unique reproducible chip identities result from response variability between chips. Typical PUFs are distinct components having no functional purpose. Touchscreen current sensors exhibit PUF properties [1], [2]. Their data is therefore unique to a user-device combination [2]. Responses generated on one device are invalid on another.

a) Contributions:

Identity Mapping – Universally and naturally generated input pressure token streams are mapped to a profile. Profiles are a set of n-grams. n-grams contain n token in-fixes of the pressure token sequences with the outgoing transition probability to each token in the alphabet. Tokens are a function of soft keyboard key location and sensor current magnitude (user pressure); they capture unique user and device characteristics.

Distance Metric – Authentication utilizes a profile distance metric defined between profile n-gram sets.

II. Implementation

A continuous biometric authentication scheme is proposed. Software keyboard touchscreen interactions provide biometric data. Data PUF properties – variability and reliability – enhance authentication security. User profiles are n-gram approximations to model Markov processes. This simplifies profile authentication computations using a distance metric. The profile data structure is optimized using a prefix tree to reduce the profile size and to accelerate profile authentication computations.

A sequence of curated pressure tokens is parsed into n-grams: a sequence of n tokens followed by the probability of the $(n+1)$st token over the entire alphabet. This collection of n-grams from a sequence of touchscreen interactions is a user profile.

Software keyboard interactions generate data. Data tokenization generates a token sequence. An n-gram model predicts next token probabilities given this sequence. Authentication uses a distance metric to compare profiles.

a) Data: We modified a software keyboard application to collect `MotionEvents`. Four users completed at least 5000 interactions on 3 Nexus 7 tablets.

b) Input Tokenization: Tokens contain (x, y) location ranges to identify the spatial span of an individual key in the soft keyboard. Pressure ranges divide each location range into *Token Number* tokens. Pressure ranges depend on location pressure distributions computed for each location range. *Token Number* tokens in range $\mu \pm 2\sigma$ are created having equal numerical size. $\mu = 0.5$ with $\sigma = 0.1$ with *Token Number*$= 4$ results in ranges $[0.3, 0.4), [0.4, 0.5), [0.5, 0.6), [0.6, 0.7]$. $\mu \pm 2\sigma$ contains

2380-6923/19/$31.00 ©2019 IEEE
DOI 10.1109/VLSID.2019.00125

95.45% of profile pressure values [3]. Choosing this pressure range excludes outliers increasing reproducibility.

Tokenization maps raw `MotionEvents` to these model tokens. n-grams are token sequences.

c) Next Token Probabilities: A token, T_i, succeeds each n-gram, W_j. Token sequence T_k to T_l occurs $O(T_k, \ldots, T_l)$ times. Next token probability for T_i occurring after W_j is $P(T_i|W_j)$; this is number of occurrences of T_i after W_j divided by $O(W_j)$.

d) Distance Metric: User profiles are sets of n-grams. An n-gram NG consists of a window of n token sequence W along with a successor frequency vector v represented as $NG(W, v)$. The successor frequency vector captures the transition frequency from W to all possible tokens in the token alphabet $\mathcal{T}$ – the result of tokenization described earlier. $NG.W$ ($NG.v$) refers to the window (vector) component of the n-gram NG. Equation 1 defines a profile distance; it is a weighted sum of next token probability differences.

$D_{profile}(B, P) :=$ Distance between profiles B and P.

$D_{n-gram}(NG1, NG2) :=$ Distance between n-grams $NG1 = (W1, v1)$ and $NG2 = (W2, v2)$.

We define a search function $NG(P, W)$ to denote the n-gram with window W in profile P. If a window W n-gram does not exist in profile P, this function returns null n-gram with vector $v = [0, 0, \ldots, 0]$.

Profile B is authenticated against user profile P. For each n-gram NG_i^B in B, $D_{profile}$ takes the distance between NG_i^B and its twin n-gram in profile P given by $NG(P, NG_i^B.W)$. This distance between n-grams, $D_{n-gram}(NG_i^B, NG(P, NG_i^B.W))$, is further weighted by the frequency of $NG_i^B.W$ in B. Note that this weight selection from profile B makes this distance metric asymmetric.

$D_{n-gram}(NG1, NG2)$ computes the distance over all transitions or entries in the vector $NG1.v$, which is computed as $|(NG1.v[i] - NG2.v[i])|$ which is further weighted by the frequency of the transition on token T_i leading to $|(NG1.v[i] - NG2.v[i])| * NG1.v[i]$. This expression is summed up over each token $T_i \in \mathcal{T}$.

$$
\begin{aligned}
&D_{profile}(B, P) = \\
&\Sigma_{i=0}^{|B|-1}\Big(D_{n-gram}(NG_i^B, NG(P, NG_i^B.W)) * \frac{O(NG_i^B.W)}{|B|}\Big) \\
&D_{n-gram}(NG1, NG2) = \\
&\Sigma_{i=0}^{|\mathcal{T}|-1}\Big(|NG1.v[i] - NG2.v[i]| * NG1.v[i]\Big)
\end{aligned}
\tag{1}
$$

The distance metric ranges from 0.0 to 1.0; it describes how much the interaction profile B deviates from user profile P. $D_{profile} = 0.0$ is maximally close.

Order matters – $D_{profile}(P, B) \neq D_{profile}(B, P)$. The former compares n-grams in B to equivalent n-grams in P. Comparing n-grams in P not contained in B says nothing about B; therefore it is avoided.

Authentication, $AUTH(P, B)$, compares an interaction profile B against a user profile P. Authentication compares $D_{profile}(P, B)$ against a threshold.

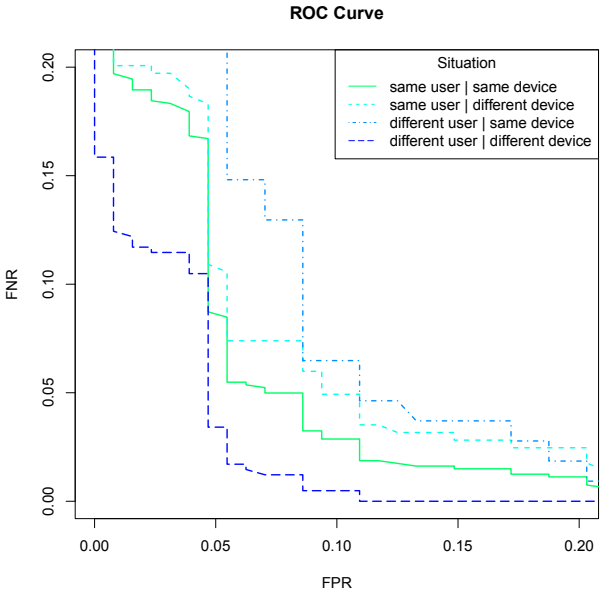

Fig. 1. Classification is performed in four situations. *Total Model Size* 4800 achieves 78.7% accuracy exposing relative situation accuracy differences. Points closest to the bottom-left represent lowest error.

III. Results

a) Authentication: Quantifying error rates describes operating characteristics. Receiver Operating Characteristic (ROC) curves relate error rates; this is interpreted as trade-offs in system behavior. Varying *Authentication Threshold* generates curve points.

FNR and FPR error rates are quantified. Figure 1 provides scenario ROC curves. *Total Model Size* 4800 achieves 78.7% accuracy exposing relative situation accuracy differences. Scenario lines compare a scenario to a same user, same device profile. The same user, same device line compares the same user, same device scenario against all other scenarios; it quantifies aggregate system performance. The different user, different device scenario has highest accuracy. This is expected. It indicates the profile difference metric is a function of both user and device uniqueness. This confirms user and device characteristics contribute to the computation.

b) Computation Time: An Android program evaluates classification time on a Nexus 7 tablet. Computation time increases with *Total Model Size* requiring 1 second per 3333.33 interactions. Accuracy of 80% is achievable in under 2 seconds – 90% in under 3 seconds – 100% in under 4 seconds.

Acknowledgements: This project was supported by the Dept. of Homeland Security, Science and Technology Directorate under Contract # D 1 5PC00158 and by NSF Grant CNS 1441640.

References

[1] Kurt Rosenfeld, Efstratios Gavas, and Ramesh Karri. Sensor physical unclonable functions. In *Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on*, pages 112–117. IEEE, 2010.

[2] R. Scheel and A. Tyagi. Characterizing composite user-device touchscreen physical unclonable functions (pufs) for mobile device authentication. In *ACM International Workshop in Trusted Embedded Devices, TRUSTED 2015*. ACM, October 2015.

[3] Friedrich Pukelsheim. The Three Sigma Rule. *The American Statistician*, 48(2):88–91, May 1994.