Continuous Authentication on Mobile Devices by Analysis of Typing Motion Behavior

Hugo Gascon[1], Sebastian Uellenbeck[2], Christopher Wolf[2], Konrad Rieck[1]

[1]Computer Security Group
University of Göttingen
Goldschmidtstrasse 7
37073 Göttingen
{hgascon, konrad.rieck}@uni-goettingen.de

[2]Horst Görtz Institute for IT-Security
Ruhr-University Bochum
Universitätsstrasse 150
44810 Bochum
{sebastian.uellenbeck, christopher.wolf}@rub.de

Abstract: Smartphones have become the standard personal device to store private or sensitive information. Widely used as every day gadget, however, they are susceptible to get lost or stolen. To protect information on a smartphone from being physically accessed by attackers, a lot of authentication methods have been proposed in recent years. Each one of them suffers from certain drawbacks, either they are easy to circumvent, vulnerable against shoulder surfing attacks, or cumbersome to use. In this paper, we present an alternative approach for user authentication that is based on the smartphone's sensors. By making use of the user's biometrical behavior while entering text into the smartphone, we transparently authenticate the user in an ongoing-fashion. In a field study, we asked more than 300 participants to enter some short sentences into a smartphone while all available sensor events were recorded to determine a typing motion fingerprint of the user. After the proper feature extraction, a machine learning classifier based on Support Vector Machines (SVM) is used to identify the authorized user. The results of our study are twofold: While our approach is able to continuously authenticate some users with high precision, there also exist participants for which no accurate motion fingerprint can be learned. We analyze these difference in detail and provide guidelines for similar problems.

1 Introduction

Smartphones have become the epitome of the *always on, always connected* trend. They combine and extend the functionality of feature phones with a multitude of characteristics from desktop computers, and this have made them a world-wide commodity with an ever increasing market share. Unfortunately, the great amount of personal data stored on these devices, including passwords or banking information, have made them an attractive target

for criminals. To prevent this information from being physically retrieved by attackers, access control mechanisms like user authentication have been proposed. When focusing on the Android OS, user authentication can mainly be divided into two categories.

First, methods that are based on knowledge like PIN, password, and Android Unlock Patterns. Second, biometrics based methods. It is well known that knowledge-based authentication factors, such as passwords and PINs, provide only limited security on smartphones. Similarly, some biometric factors, such as Face Unlock, also fail to provide a reliable user authentication (c. f., [MDAB10, BPA12, UDWH13, FM12]). In either case: A smartphone that was unlocked once stays unlocked until it is actively locked again or a fixed period of time elapses without interaction. Therefore, there always exists a time frame in which an attacker can steal the unlocked phone to later obtain all data stored on the device.

Our Contribution In this paper, we present a user authentication approach that is based on analyzing the typing motion behavior of a specific user. To counter attacks that happen after the smartphone is unlocked, our approach uses a continuous authentication scheme by means of biometry while the user is entering text. In general, biometric features can be divided into the two classes: physiological and behavioral biometrics. Physiological biometrics rely on something *the user is*. This class contains features like fingerprint [CKL03], face [ZCPR03], hand geometry, voice, and iris. They are often used for authentication to high secure entrance systems [JRP06] but need special hardware to be identifiable. Apart from that, behavioral biometrics are based on how *the user behaves*, like keystroke patterns, the user's gait [GS09], or location information without requiring additional hardware. This class can be used to authenticate the user uninterruptedly and is also implementable with built-in smartphone sensors.

To gather behavioral biometrics, we have developed a software keyboard application for the Android OS that stores all sensor data for further learning and evaluation. In a field study, we asked more than 300 participants to enter a short text into our smartphone. To ensure that only motion data related to typing behavior is used to learn a profile, we preprocessed the sensor signals according to certain time constraints. Next, we extracted various features leading to a 2376-dimensional vector representing the typing motion behavior of a user in a given time frame. By means of SVMs, we learn a classifier in order to identify the behavioral fingerprint of each user. In the end, we analyze the results of our approach and its limitations, and discuss what obstacles can be found when tackling similar problems and their possible improvements.

Summary In summary, we make the following contributions:

- We implemented a software keyboard for Android that enabled us to collect the typing behavior of 300 users in a field study.
- We designed a time-based feature extraction method and evaluated the performance of a machine learning classifier in a continuous authentication problem.
- Finally, we discuss limitations to sensor-based approaches in authentication problems and propose several improvements and considerations.

2 Related Work

Besides classic user authentication on smartphones by means of PINs or passwords, there has been a lot of work in this area in recent years. Frank et al. [FBM+13] evaluated the feasibility of continuous user authentication by means of touchscreen features. Although they confirmed that user authentication is possible despite having a False Negative Rate of only 0% to 4%, they do not consider other features obtained by motion or position sensors.

Zhen et al. [ZBHW12] proposed an extension to the plain PIN authentication method strengthened with sensor data and timings. They collected five different acceleration values, the touching pressure, the touched area on the screen, and different time values like key-hold time or inter-key time. They evaluated previously specified 4-digit and 8-digit PINs from over 80 participants. As a result they achieve an equal error rate between 3.65% and 7.34% referred to the predefined PIN. In contrast to them, we propose a continuous authentication methodology intended to be used without taking predefined text into consideration. After a learning phase, users are authenticated while entering normal text.

Sandnes and Zhang [SZ12] studied strategies for identifying users based on touch dynamics. They monitor and extract features like left vs. right hand dominance, one-handed vs. bimanual operation, stroke size, stroke timing, symmetry, stroke speed, and timing regularity. In an experiment with 20 participants, they found their approach and feature set useful. In contrast to us, they did not consider the behavior of attackers and had a very limited set of participants. They could therefore not give any numbers on how this approach performs in reality. Shi et al. [SNJC10] investigated implicit authentication through learning the behavior of a user. They created user patterns based on sent and received text messages and phone calls, the browser history, and the location of the smartphone. This approach also allowed for considering typing behavior whereas this was only a theoretic but not reviewed aspect of their work.

De Luca et al. [DHB+12] used biometric features to improve the security of the Android Unlock screen. While performing the wipe gesture to unlock the smartphone's screen, they collected all data available from the touchscreen, including pressure, size, coordinates, and time. In a study with 48 participants, they could show that their approach reaches a 98% True Positive Rate but comes also with the price of a 43% False Positive Rate for the best case.

Li et al. [LZX13] proposed a system similar to our approach, that uses a continuous authentication for smartphones by taking the user's finger movement pattern into account for learning. In contrast to us, they did not consider entering text into a softkeyboard but only gestures like sliding towards a special direction or taps.

Keystroke dynamics have also been considered for desktop computers [MR00, PKW04, UW85, ASL+04]. Since smartphones comprise sensors to capture environmental changes, they offer more capabilities to authenticate users. In the following, we show that sensor data can be used to gather typing motion behavior.

3 Methodology

In this section we describe the different steps of our method. First, we present how the user data is collected by means of our softkeyboard prototype and then the feature building process is described. These features allow us to design a learning and classification setup that lets us identify a user during the ongoing interaction with the device.

3.1 Data Collection

Smartphones comprise many sensors that can be used to evaluate real-world characteristics, like user biometrics. Still they have to been processed through adequate software beforehand. The majority of smartphones do not contain hardware keyboards to enter characters but a touchscreen that can be utilized as keyboard. Here, software displays a virtual keyboard (softkeyboard) on the touchscreen and the user's input is forwarded by the touchscreen to the underling application.

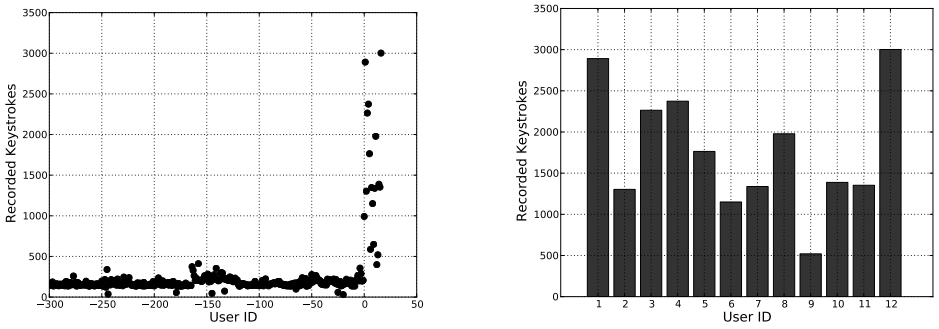
In this paper, our goal is to explore the possibility of learning an individual profile from the motion behavior of a smartphone while the user is typing. This model will then be used to continuously authenticate the user against unauthorized attackers. To this aim and in order to collect the user's motion data, we have developed a softkeyboard prototype for the Android OS being able to read and store all sensor events for further analysis.

The softkeyboard makes use of the accelerometer to measure acceleration, the gyroscope to measure torque and the orientation sensor to measure the relative position of the device. Each one of these parameters is measured in the three dimensions of space x, y, and z. Additionally, our prototype utilizes the display to get the exact position the user has touched to introduce a keystroke. On the software-side, a keystroke is divided into three events: *onPress*, *onKey*, and *onRelease*. The first is raised when the user touches the screen, the second when the software sends the character to the underlaying layer, and the last presents the event when the user's finger leaves the screen. This fine-grained approach is necessary since smartphones allow to change the selected character after the *onPress* event so the final character is only fixed when the user's finger releases the screen. For each of this three events, we record the timestamp with a granularity of milliseconds and all sensor values. Since the duration of a keystroke can—depending on the user—last from 80 ms to 500 ms, we record all sensor events that occur while a user is entering text.

In order to analyze the typing motion behavior, we asked 315 people to write a short and predefined text ($\approx$ 160 characters) on the test smartphone using our softkeyboard prototype. The text introduced by the subjects was a set of different *pangrams* containing all letters of the English language like *The quick brown fox jumps over the lazy dog*. By choosing such sentences we assured that the users had to touch each virtual character at least once. When asking them to support our work we explained that their data was taken and recorded for a scientific experiment.

We split the keystroke data into two fractions. The first fraction was taken from 303 participants that entered the text only once. This group represents individuals who in a figurative attack scenario, take the device away while unlocked. Their typing motion behavior is

therefore considered as an attempt to interact with the device unauthorized. The second fraction, the remaining 12 participants, were asked more than 10 times to write the same short text. This group was intended to be considered as authorized users. Their larger amount of data should allow us to model a unique profile that can be used to individually distinguish their behavior from the one of any other unauthorized attacker. Figure 1(a) shows the total number of keystrokes introduced by each user. In the following section we present the preprocessing steps to extract relevant features from the sensor signals recorded in real time and which will be fed to the classifier.



(a) Keystrokes per User

(b) Keystrokes per Authorized User

Figure 1: Total number of keystrokes introduced per unauthorized and authorized users.

3.2 Time-based Feature Extraction

During the data collection phase, different users introduce text through the softkeyboard while motion data is continuously recorded by the sensors. In order to ensure that only motion information related to typing behavior is used to learn each user profile, the sensor signals need to be processed according to certain time constraints.

The goal during the feature extraction phase is to compute a data point from all values acquired during T seconds while a user is typing. Figure 2 shows how different timers are used to correctly identify and record the data captured by the sensors.

Once a user has started typing, a data point is computed every T seconds. If the user does not introduce any keystroke during T_{stop} seconds, we assume that the user has stopped typing. This allows us to discard sensor values measured after T_1. In case that recorded values do not sum up to T seconds and the user resumes typing, new sensor values will be added to the previously captured up to complete the T seconds. Since data points are computed on a time basis, the burst of keystrokes used to compute each data point may be different and depends on the typing speed and pauses performed by the user.

Captured signals are then normalized to remove the artifacts or constant values such as the $9.81\ m/s^2$ typically measured in the accelerometer z coordinate as result of gravity. The normalization is performed in a similar manner as described by Aviv et al. [ASBS12].

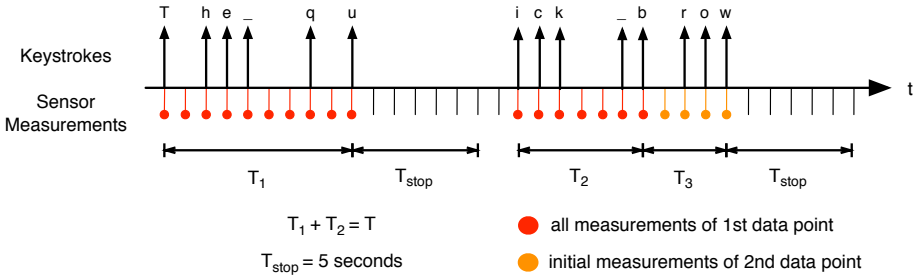


Figure 2: Description of the relation between typed keystrokes and motion measurements. Certain timing conventions are introduced to slice sensor signals according to a predefined time window.

As mentioned above, the 9 signals resulting from the spatial readings of the accelerometer, gyroscope, and orientation sensors are recorded during T seconds. Three normalized forms are computed for each one of them. Table 1 shows how a mean, linear, and 3-degree spline (P_{3d}) are subtracted from the original values to obtain the three normalized versions that will be used to extract a set of discrete features.

Table 1: Description of the 3 normalization forms applied to each sensor signal.

Type	**Fit**	**Normalization**
Mean	$m = \frac{1}{N}\sum_1^N s_i$	$S_m = S - m$
Linear	$mx + c$	$S_l = S - (mx + c)$
Spline	P_{3d}	$S_p = S - P_{3d}$

Figure 3 presents a visual example of one of the sensor signals. The three types of fit reconstructions are over imposed to the original signal. Their respective subtraction in each one of the cases will produce the normalized signals S_m, S_l and S_P. The set of features described in Table 2 are then extracted from each sequence of normalized values.

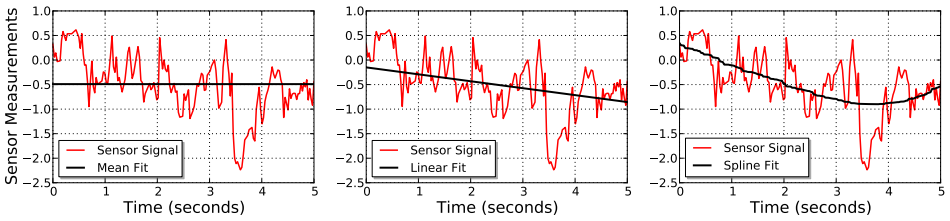


Figure 3: Example of accelerometer sensor readings for T = 5 in the x direction and the corresponding three types of fit reconstructions.

Table 2: Set of features extracted from each one of the normalized sensor signals.

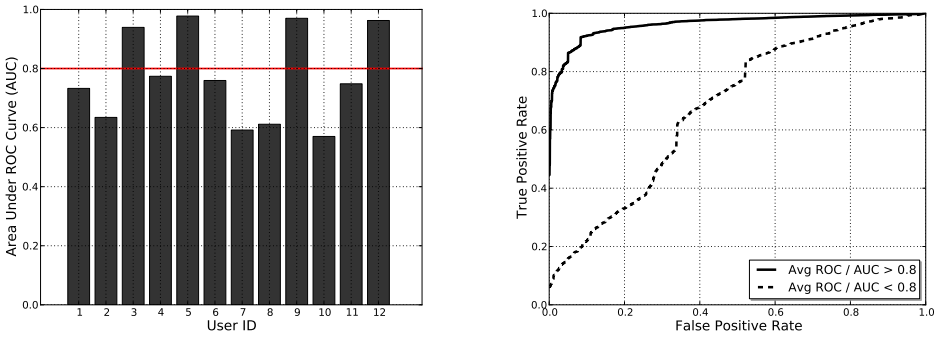
Features	Length	Description
Simple Statistics	6	root mean square, mean, standard deviation, variance, max and min
Spline Coefficients	6	coefficients of a 5-degree smoothing spline
Spline Simple Statistics	6	simple statistics from 5-degree smoothing spline
iFFT Spline Features	35	inverse DFT of a DFT of the 5-d spline fit curve using 35 samples
iFFT Signal Features	35	inverse DFT of a DFT of the sensor signal using 35 samples

The total number of features extracted from each normalized signal is therefore 88. As the signals from 9 sensors are recorded simultaneously and 3 normalize versions are reconstructed, the total number of features computed is $3 \times 9 \times 88 = 2376$. As a result, a 2376-dimensional vector, representing a unique data point, is computed every T seconds when the user is typing.

4 Evaluation

Our aim is to be able to correctly identify the profile of an authorized user versus that of several unauthorized attackers. Therefore, we pose a two-class classification problem. The first class, labeled $+1$, is formed by the data points generated from the signals of one of the 12 users with the largest amount of recorded data. The second class, labeled -1, is formed by the data points generated from the typing behavior of the 303 users that have introduced a short sequence of characters. A linear Support Vector Machine (SVM) is finally used to find the hyperplane that optimally splits both classes of data points in the 2376-dimensional space spanned by the generated featured vectors.

To evaluate the classification performance, we compute the Receiver Operating Characteristics (ROC) curve for the classifier of each user. The ROC curve presents the performance of the classifier in terms of *true positive rate* (TPR) versus *false positive rate* (FPR), where the area under the curve (AUC) of a perfect classifier has a value of 1. We can observe in Figure 4(a) that the classifiers learnt for the different users show a heterogeneous performance. While users with ID 3, 5, 9 and 12 can be identified with an AUC above 0.9, the AUC remains under 0.6 for some of the other users. These results suggest that only a certain number of users have a distinct and characteristic profile. We have established that a user requires a classification AUC above 0.8 to be identified with sufficient confidence. Now we can distinguish between identifiable and non-identifiable users. Figure 4(b) presents the average ROC for all of the users in both groups. The average classification performance of the classifiers for non-identifiable users presents a considerable high FPR of 35%, while reaching only a TPR of 58% at this point. On the contrary, if we consider the group of clearly identifiable users, the average classification performance achieves a TPR of 92%, reached at only 1% of FPR.



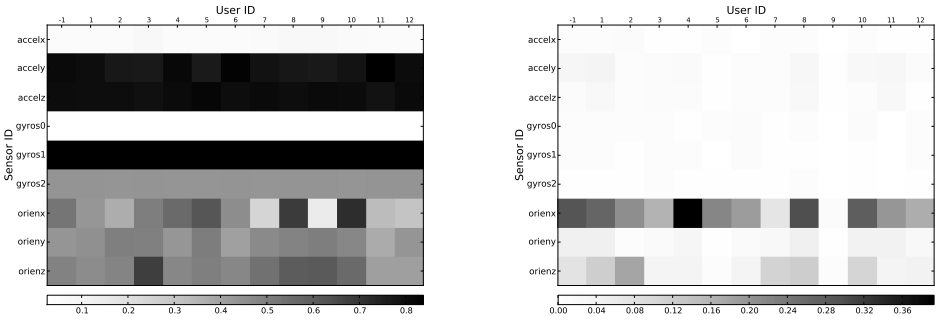
(a) AUC per user

(b) Averaged ROCs

Figure 4: Classification performance for each authorized user. Figure 4(b) shows both, the average ROC of users with an individual AUC over 0.8 and the average ROC of users with an individual AUC under 0.8.

5 Discussion

In order to explain such an uneven performance and also to see what makes a user identifiable, we observe the original sensor measurements for each user. Figure 5 presents the sample mean and standard deviation of the normalized values recorded by each sensor. It can be noticed that values recorded by the accelerometer and gyroscope sensors have very similar average values with a very small dispersion for all the users, including the class of unauthorized users labeled -1. In fact, we see that the largest differences between the typing motion behavior of the users is mainly characterized by the orientation sensor. In particular, the average from the values recorded in the x coordinate presents not only the higher differentiation between users but also a higher and disparate dispersion.



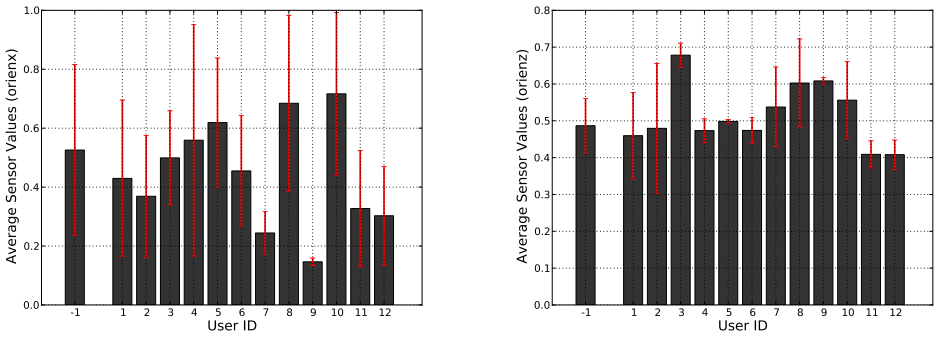
(a) Mean of sensor values

(b) Standard deviation of sensor values

Figure 5: Mean and standard deviation of the sensor values recorded from each user and normalized between 0 and 1. User ID -1 represents all measurements corresponding to users with negative IDs (attackers).

It should be noted that although the mean and standard deviation of the sensor values can provide certain information about the dataset and the user profiles, the SVM is able to determine more complex relationships within the data in the high-dimensional feature space built from the original sensor values. This may be difficult to explain but led to better results. Nevertheless, these simple statistics already allow us to draw some clear conclusions on the good classification performance of the identifiable users.

For example, Figure 6(a) shows how user 9 presents a very distinctive behavior with the lowest mean value and a extremely low standard deviation in the x coordinate of the orientation sensor. This is very interesting as this user have introduced a smaller amount of data. In the same way, Figure 6(b) shows how user 3 presents a very high mean value on the z coordinate of the orientation sensor, while having also a very low standard deviation. We thus conclude that the high variance on the x coordinate of the orientation is a root-cause for the differences in authentication performance. Likely, some users seem to randomly shake or shift the smartphone during typing on this particular axis and thereby limit the learning of an accurate behavioral fingerprint.



(a) Orientation sensor, x coordinate

(b) Orientation sensor, z coordinate

Figure 6: Mean and standard deviation of measurements recorded by the accelerometer in the x and z coordinates for all users.

In spite of the uneven performance, there exist some considerations that we have taken into account to assure that the obtained results are sound, and also some remarks that need to be addressed by any research dealing with similar problems to the one described in this paper. In the first place, in every experiment that measures human behavior and therefore requires human test subjects, the environmental conditions during the collection phase need to be extremely well controlled. As reported in previous research, even the behavior of a unique person can greatly differ from one day to another. Consequently, not only different measurements from the same person have been taken on different days, but in a well conditioned setting in order to minimize the influence of the environment in the captured data.

Additionally, a particular issue may arise when dealing with time series. In order to select the best parameters for the learnt model during the training phase a typical cross-validation strategy can be followed. However, the time dependence of the data imposes certain con-

straints during the phases of training and validation which are often overlooked. For instance, in a k-fold cross validation, the data points in each fold should not be randomized. This may improve the final performance but would also be an unrealistic setup if learning is done on a real device while the user is introducing text. If we consider each sequence of characters as independent from each other, cross-validation folds may be shuffled as blocks but the data points that form each one of them should never be randomized. Furthermore, the data points used for testing need to lie in the future of the training and validation sets for the same reason. Moreover, the non-trivial problem of selecting the best possible parameters to model each user can led to a decrease in performance. An extensive analysis of the influence of parameters like T_{stop}, the degree for spline reconstructions of the signals or the length of the different FFT could reveal better combinations of values to identify each individual user.

6 Conclusions and Future Work

In recent years, many different solutions to security problems but also attacks involving the use of smartphone sensors have been proposed. The standard adoption of touchscreens in these devices has led to a new strain of biometric related research, aiming in most cases at improving the traditional authentication scheme of text-based passwords. As the continuous interaction of users with these devices allows for a continuous analysis of their behavior, we have explored in this paper how the motion information retrieved by the sensors can be used to build a unique typing motion profile of an authorized user. During the evaluation and discussion sections several conclusions have been drawn regarding the achieved performance. We have shown that certain users show a characteristic behavior which can be identified with a TPR of 92% at a FPR of 1%, while some other users are hardly distinguishable from each other and from the typing motion behavior of the attackers.

These results and the proposed method open the door to future improvements. For instance, we have posed the problem as a binary classification experiment, however other strategies may come to mind. In the general case or if attacker data is scarce, an anomaly detection setup can be implemented, where the authorized user is modeled using a one-class SVM. Moreover, in this type of problem, the false positive rate should be minimized to avoid constant lockings of the device and additional requests to re-authorize the user with text or graphical passwords. Certain techniques as majority voting or making the classifier decide that a data point belongs to an unauthorized user only after a number of data points have been identified as an attack, are interesting paths which we plan to explore in our future work.

References

[ASBS12] Adam J. Aviv, Benjamin Sapp, Matt Blaze, and Jonathan M. Smith. Practicality of Accelerometer Side Channels on Smartphones. In Robert H'obbes' Zakon, editor, *ACSAC*, pages 41–50. ACM, 2012.

[ASL+04] Lívia C. F. Araújo, Luiz H. R. Sucupira, Miguel Gustavo Lizárraga, Lee Luan Ling, and João Baptista T. Yabu-uti. User Authentication through Typing Biometrics Features. In David Zhang and Anil K. Jain, editors, *ICBA*, volume 3072 of *Lecture Notes in Computer Science*, pages 694–700. Springer, 2004.

[BPA12] Joseph Bonneau, Sören Preibusch, and Ross Anderson. A Birthday Present Every Eleven Wallets? The Security of Customer-Chosen Banking PINs. In Angelos D. Keromytis, editor, *Financial Cryptography*, volume 7397 of *Lecture Notes in Computer Science*, pages 25–40. Springer, 2012.

[CKL03] T. Charles Clancy, Negar Kiyavash, and Dennis J. Lin. Secure Smartcard-based Fingerprint Authentication. In *ACM SIGMM Workshop on Biometric Methods and Applications*, pages 45–52, 2003.

[DHB+12] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. Touch Me Once and I Know it's You! Implicit Authentication Based on Touch Screen Patterns. In Joseph A. Konstan, Ed H. Chi, and Kristina Höök, editors, *CHI*, pages 987–996. ACM, 2012.

[FBM+13] Mario Frank, Ralf Biedert, Eugene Ma, Ivan Martinovic, and Dawn Song. Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication. *IEEE Transactions on Information Forensics and Security*, 8(1):136–148, 2013.

[FM12] Rainhard Dieter Findling and Rene Mayrhofer. Towards Face Unlock: On the Difficulty of Reliably Detecting Faces on Mobile Phones. In Eric Pardede and David Taniar, editors, *MoMM*, pages 275–280. ACM, 2012.

[GS09] Davrondzhon Gafurov and Einar Snekkenes. Gait Recognition Using Wearable Motion Recording Sensors. *EURASIP J. Adv. Sig. Proc.*, 2009, 2009.

[JRP06] Anil K. Jain, Arun Ross, and Sharath Pankanti. Biometrics: A Tool for Information Security. *IEEE Transactions on Information Forensics and Security*, 1(2):125–143, 2006.

[LZX13] Lingjun Li, Xinxin Zhao, and Guoliang Xue. Unobservable Re-authentication for Smartphones. In *NDSS*. The Internet Society, 2013.

[MDAB10] Steven J. Murdoch, Saar Drimer, Ross J. Anderson, and Mike Bond. Chip and PIN is Broken. In *IEEE Symposium on Security and Privacy*, pages 433–446. IEEE Computer Society, 2010.

[MR00] Fabian Monrose and Aviel D. Rubin. Keystroke Dynamics as a Biometric for Authentication. *Future Generation Comp. Syst.*, 16(4):351–359, 2000.

[PKW04] Alen Peacock, Xian Ke, and Matthew Wilkerson. Typing Patterns: A Key to User Identification. *IEEE Security & Privacy*, 2(5):40–47, 2004.

[SNJC10] Elaine Shi, Yuan Niu, Markus Jakobsson, and Richard Chow. Implicit Authentication through Learning User Behavior. In Mike Burmester, Gene Tsudik, Spyros S. Magliveras, and Ivana Ilic, editors, *ISC*, volume 6531 of *Lecture Notes in Computer Science*, pages 99–113. Springer, 2010.

[SZ12] Frode Eika Sandnes and Xiaoli Zhang. User Identification Based on Touch Dynamics. In Bernady O. Apduhan, Ching-Hsien Hsu, Tadashi Dohi, Kenji Ishida, Laurence Tianruo Yang, and Jianhua Ma, editors, *UIC/ATC*, pages 256–263. IEEE Computer Society, 2012.

[UDWH13] Sebastian Uellenbeck, Markus Dürmuth, Christopher Wolf, and Thorsten Holz. Quantifying the Security of Graphical Passwords: The Case of Android Unlock Patterns. In Ahmad-Reza Sadeghi, Virgil D. Gligor, and Moti Yung, editors, *ACM Conference on Computer and Communications Security*, pages 161–172. ACM, 2013.

[UW85] David A. Umphress and Glen Williams. Identity Verification Through Keyboard Characteristics. *International Journal of Man-Machine Studies*, 23(3):263–273, 1985.

[ZBHW12] Nan Zheng, Kun Bai, Hai Huang, and Haining Wang. You Are How You Touch: User Verification on Smartphones via Tapping Behaviour. Technical report, College of William & Mary, Williamsburg, VA, USA, December 2012.

[ZCPR03] Wen-Yi Zhao, Rama Chellappa, P. Jonathon Phillips, and Azriel Rosenfeld. Face Recognition: A Literature Survey. *ACM Comput. Surv.*, 35(4):399–458, 2003.