I Sensed It Was You: Authenticating Mobile Users with Sensor-Enhanced Keystroke Dynamics

Cristiano Giuffrida[1], Kamil Majdanik[1], Mauro Conti[2], and Herbert Bos[1]

[1] VU University Amsterdam, The Netherlands
{giuffrida,k.majdanik,herbertb}@cs.vu.nl
[2] University of Padua, Italy
conti@math.unipd.it

Abstract. Mobile devices have become an important part of our everyday life, harvesting more and more confidential user information. Their portable nature and the great exposure to security attacks, however, call out for stronger authentication mechanisms than simple password-based identification. Biometric authentication techniques have shown potential in this context. Unfortunately, prior approaches are either excessively prone to forgery or have too low accuracy to foster widespread adoption.

In this paper, we propose *sensor-enhanced keystroke dynamics*, a new biometric mechanism to authenticate users typing on mobile devices. The key idea is to characterize the typing behavior of the user via unique sensor features and rely on standard machine learning techniques to perform user authentication. To demonstrate the effectiveness of our approach, we implemented an Android prototype system termed Unagi. Our implementation supports several feature extraction and detection algorithms for evaluation and comparison purposes. Experimental results demonstrate that sensor-enhanced keystroke dynamics can improve the accuracy of recent gestured-based authentication mechanisms (i.e., $EER > 0.5\%$) by one order of magnitude, and the accuracy of traditional keystroke dynamics (i.e., $EER > 7\%$) by two orders of magnitude.

1 Introduction

Recent years have witnessed the blossom of the mobile computing era, with a sharp increase in the number of handheld devices and mobile users. According to [1], the number of mobile-connected devices exceeded the number of people on earth at the end of 2013, with projections indicating a steady increase in the next few years. The pervasive nature of these devices and their increasingly enhanced computing power and storage capacity has created opportunities for many growingly popular mobile services, ranging from email and photo sharing to financial services such as e-commerce and mobile banking.

As our everyday reliance on mobile services increases, so does the amount of sensitive information harvested in handheld devices, such as passwords and credit card numbers. Adequately protecting such private data from unauthorized access

S. Dietrich (Ed.): DIMVA 2014, LNCS 8550, pp. 92–111, 2014.

is an increasingly pressing concern, also given the small and portable nature of mobile devices and their great exposure to prying eyes. For instance, smartphone theft affected 1.6 million devices in 2012 in the U.S. alone [3]—with the majority of finders [2] attempting to access private user data.

Unfortunately, traditional password-based (or PIN- or pattern-based) authentication schemes commonly used on mobile devices have a number of weaknesses that can inadvertently expose the user to security breaches. First, they are susceptible to guessing attacks, with as many as 91% of the passwords found in the top 1000 list [9], a problem exacerbated by the constrained nature of mobile devices that encourages users to select simpler and weaker passwords. Second, they are susceptible to smudge attacks, where attackers infer passwords from the finger smudges left on the touch screen [5]. Finally, they are susceptible to shoulder-surfing attacks [54], where attackers rely on direct observation to steal passwords in a public setting. Recent attacks have also become automated and more sophisticated, with attackers stealing passwords using low-end cameras and fingertip motion analysis through repeated reflections [58].

Interestingly, studies have shown that users are generally favorable to alternative authentication mechanisms [15], which has spurred research on biometric authentication for mobile devices. Several schemes have been proposed in recent years, such as identifying users based on their gaits [37], shake motions [43], phone-to-ear gestures [16], touch gestures [18,19,33,39,51], or keystroke dynamics [23,55,56].

While these approaches have shown potential, they generally yield unacceptably low accuracy to foster widespread adoption. In fact, the equal error rates ($EERs$) of such approaches are typically greater than 5% or even 10%. A notable exception is given by recent work on touch gesture-based authentication [51], which reported $EERs$ of as low as 0.5% using a fine-grained stroke characterization strategy. Gesture-based schemes, however, have been shown extremely vulnerable to simple statistical attacks. While relying only on general population statistics, such attacks can easily yield a substantial EER increase (between +35.14% and +44.07%) [50]. Keystroke dynamics [29], in contrast, has been shown robust against human [28] and synthetic [53] attacks—although more recent studies seem to suggest a small EER increase (between +3.8% and +7.6%) [49]–and attacks that have been shown to yield substantial EER increases are only possible with access to the set of the victim's typing patterns obtained from an implanted keylogger [38,45]. Unfortunately, traditional keystroke dynamics techniques are also plagued by low accuracy ($EER > 7\%$) [23,28].

In this paper, we present *sensor-enhanced keystroke dynamics*, a new authentication mechanism for sensor-equipped mobile devices with a touch screen and a software keyboard. The key idea is to combine the traditional timing-based characterization adopted in keystroke dynamics with movement sensors information that reflects the unique typing behavior of each user, while relying on standard machine learning techniques to perform authentication. The richer feature set aims to substantially improve the accuracy of prior approaches and also enhance the robustness against human or synthetic attacks. Unlike prior

attempts to enrich keystroke dynamics with nonconventional features [47, 55], our feature extraction strategy relies on timing-agnostic metrics computed over a sliding window to describe a given sensor-sampled distribution. This strategy is crucial to perform high-accuracy user identification, outperforming all the prior biometric authentication mechanisms for mobile devices.

Contribution. The contribution of this paper is threefold:

- First, we introduce *sensor-enhanced keystroke dynamics*, a new technique to authenticate users typing on a mobile device via keystroke timings—akin to traditional keystroke dynamics—and movement sensor information—i.e., information from accelerometer and gyroscope.
- Second, we implemented UNAGI, a fixed-text authentication system based on sensor-enhanced keystroke dynamics for Android. While sensor-enhanced keystroke dynamics can be also used in free-text authentication scenarios, our focus is on fixed-text—and thus static—authentication here. UNAGI supports several feature extraction and detection algorithms for evaluation purposes.
- Third, we ran a thorough evaluation of the proposed approach. In particular, we gathered data from 20 test subjects to evaluate and compare our techniques with prior work. Our experiments show that: (i) keystroke-induced movement sensor data are much more effective than keystroke timings in accurately identifying users; (ii) sensor-enhanced keystroke dynamics significantly improves the accuracy of state-of-the-art gesture-based authentication mechanisms for mobile devices ($EER > 0.5\%$) and of standard keystroke dynamics ($EER > 7\%$) by up to one and two orders of magnitude, respectively; (iii) our best-detector/password accuracy is sufficiently high ($EER = 0.08\%$) to enable the practical deployment of our techniques.

Organization. The remainder of this paper is structured as follows. Section 2 provides background information on keystroke and sensor dynamics. Section 3 and 4 outline the components of UNAGI and present sensor-enhanced keystroke dynamics. Section 5 evaluates and compares our techniques with prior work. Finally, Section 6 surveys related work and Section 7 concludes the paper.

2 Background

This section briefly introduces the key concepts used in our techniques.

Keystroke Dynamics

Authentication schemes based on *keystroke dynamics* consider timing information associated to key-press events to characterize the behavior of users and identify distinguishing biometric features. Authentication can be performed via *fixed-text* analysis (i.e., with the user typing some predetermined text) [7, 13, 20, 25, 28, 31, 32, 36, 42, 46] or via *free-text* analysis (i.e., with the user typing freely on the keyboard) [14, 41]. Keystroke dynamics techniques have been explored for a broad range of devices, equipped with either hardware [26, 29] or software

(also called "soft") keyboards [56]—with recent work on mobile devices largely falling into the latter category [23, 55, 56].

While different classes of keyboards (i.e., hardware vs. software, numeric vs. alphabetic, etc.) typically yield very different typing characteristics and behavioral patterns, the key-press events considered for analysis are common to all the standard keystroke dynamics techniques: (i) the *key-down* (`KD`) event, i.e., the event associated to the user pressing a given key; (ii) the *key-up* (`KU`) event, i.e., the event associated to the user releasing a given key. Most feature selection strategies described in the literature [28] consider one or more possible keystroke timings associated to consecutive key-press events, e.g., `KD-KU` time and `KD-KD` time (Figure 1). Such features are then processed by a supervised detection algorithm to identify and authenticate users.

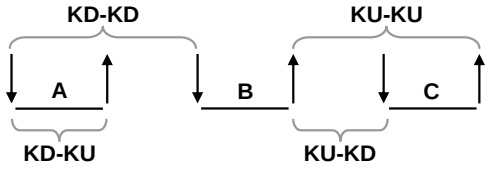

Fig. 1. Keystroke timings commonly used in keystroke dynamics techniques. The figure exemplifies the relevant keystroke events for a simple *"A-B-C"* sequence.

Sensor Dynamics

Modern mobile devices are equipped with a number of sensors that can be managed by mobile applications. The Android API, in particular, allows applications to control several different sensors, including: accelerometer, gyroscope, temperature, air pressure, gravity, light, magnetic, proximity, humidity, microphone, and camera. Our focus here is on movement sensors, that is accelerometer and gyroscope. The accelerometer measures the acceleration of the mobile device on the X (lateral), Y (longitudinal), and Z (vertical) axes. Applications can periodically sample acceleration values reported by the accelerometer. The gyroscope, in turn, measures the orientation of the device around each of the three physical axes. Applications can periodically sample orientation (angle), rate of rotation (rad/s), and rotation vector (the orientation of the device as a combination of an angle and an axis) values reported by the gyroscope.

Accelerometer and gyroscope have been extensively used in behavioral user characterization applications, as demonstrated in prior work on sensor-based keystroke [6, 10, 40, 44, 59] or location [22] inference. These techniques have successfully exploited the idea that sensor dynamics can provide very relevant information to accurately recognize the actions performed by the user on a mobile device. As an example, Figure 2 reports a sampled gyroscope distribution (y-axis) recorded with the user concurrently typing on a soft keyboard. As the figure suggests, the sensor-sampled distribution is *"perturbed"* in a systematic way every time the user issues a key-press event. Exploiting the interactions between

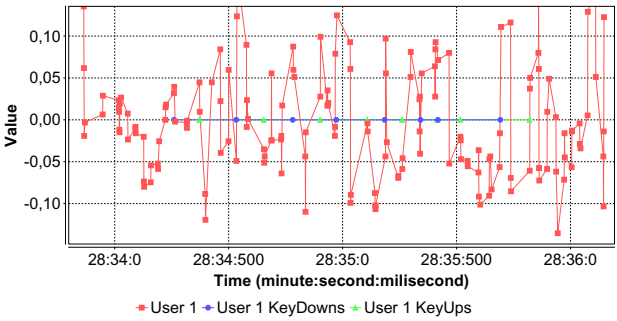

Fig. 2. Sample sensor-sampled distribution (Gyroscope, y-axis)

key-press events and the resulting *"perturbations"* induced on sensor-sampled data forms the basis for our authentication techniques.

3 Overview

Sensor-enhanced keystroke dynamics combines features from traditional keystroke dynamics techniques with features from prior sensor dynamics techniques, leveraging the unique synergies between these two classes of features on modern mobile devices. Our key intuition is to associate sensor-related data to a sequence of key-press events to improve the accuracy and robustness offered by traditional keystroke dynamics techniques. UNAGI leverages this intuition to implement a fixed-text authentication system for Android. Our current prototype is based on a modified version of the stock Android keyboard and a number of support modules that implement our sensor-enhanced keystroke dynamics techniques for authentication purposes. Figure 3 presents the high-level architecture of UNAGI.

During an authentication session (i.e., either for training or testing purposes), the user is requested to enter a fixed-text password, which is immediately processed by our authentication system for analysis. As the user interacts with the system, UNAGI intercepts (and records) all the generated key-press events and periodically samples movement sensor data from the accelerometer and the gyroscope. For this purpose, UNAGI relies on the following Android sensor sampling interfaces: `TYPE_LINEAR_ACCELERATION` and `TYPE_GYROSCOPE`. UNAGI collects sensor values at a high sampling frequency (i.e., 17Hz). This is accomplished by specifying the `SENSOR_DELAY_FASTEST` flag at sensor listener registration time.

As shown in Figure 3, all the data collected from key-press events and sensor-sampled values are processed by UNAGI's feature extraction module, which translates all the previously recorded events into features suitable for our detection algorithms. In particular, the training module processes all the features gathered during a training sessions to build—or update, in case of repetitions—a sensor-enhanced keystroke dynamics profile associated to a given user. The detection

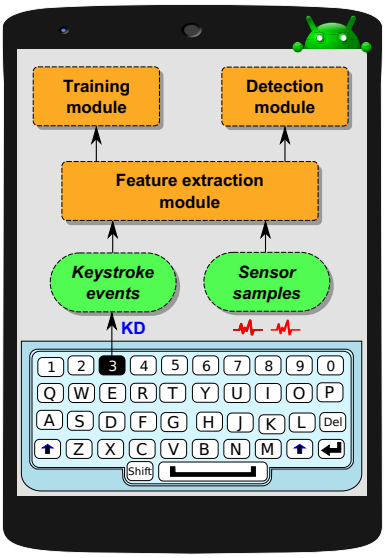

Fig. 3. Overview of UNAGI

module, in turn, matches the features gathered during a testing session against all the known user profiles to authenticate legitimate users (or detect impostors).

4 Sensor-Enhanced Keystroke Dynamics

This section details the design of our solution, with the fundamental steps required to implement a detector based on sensor-enhanced keystroke dynamics.

Data Collection

Sensor-enhanced keystroke dynamics requires different (but complementary) strategies to collect keystroke and sensor data. In particular, keystroke data are gathered as a sequence of timestamps for `KD` and `KU` events. Movement sensor data, in turn, are gathered by sampling three different distributions from the accelerometer (i.e., one distribution for each acceleration axis), and three different distributions from the orientation sensors (i.e., one for each orientation axis).

The recorded `KD` and `KU` events provide timing information only for the keys of interest. In detail, to prevent noisy measurements resulting from rarely issued key sequences, our current implementation records events only for alphanumeric characters and ignores events for all the other characters (e.g., "return" key). Sensor distributions are sampled using instantaneous sensor values provided by the Android API. A timestamp is associated to every given sample collected. For our purposes, we consider only key events issued by the user typing a predetermined password. For sensor data, we consider only samples in the time interval between 100ms before the first `KD` event and 100ms after the last `KU` event.

Feature Extraction

There are several possible strategies to extract relevant features from sensor data. As an example, Conti et al. [16] used a DTW algorithm to find similarities between two data sets. Other techniques [6, 44], rely on statistical analysis to extract relevant features from sensor data. UNAGI follows the latter approach, with features computed from a given fully typed word—or for different parts of the word—using a sliding window of predetermined size over the recorded `KD` and `KU` events. In particular, UNAGI associates features to individual unigraphs, digraphs, trigraphs, etc. (i.e., sequences of one, two, or three characters, respectively [32]). Hereafter, we use the more general term *n-graph* to refer to a sliding window of n characters defined over `KD` and `KU` events. Our notion of *n-graph* is similar to the one of *n-gram* in [8], but, in contrast to the original *n-gram* definition, we also allow nondiscrete groupings, considering, for example, *0.5*-graph intervals, as shown in Figure 4. As depicted in the figure, we allow a *0.5*-graph interval to start either on a given `KD` event and end on the next `KU` event, or start on a `KD` event and end on the next `KU` event, indiscriminately. We compute features for all the possible n-graphs using a predetermined step S ($S = 0.5$).

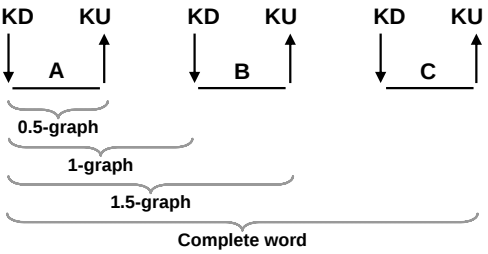

Fig. 4. Examples of n-graphs of different sizes associated to keystroke events

To select the most relevant features from the sampled sensor distributions, we rely on standard statistical metrics, a strategy inspired by existing password inference techniques [44]. In particular, UNAGI considers the following features: root mean square, minimal and maximal value, number of local maxima and minima, mean delta (mean absolute difference between two consecutive samples), sum of positive values, sum of negative values, mean value, mean value during `KU` and `KD` events, and standard deviation.

Unlike movement sensor features, extracting features associated to keystroke events is fairly established in the keystroke dynamics literature. Early keystroke dynamics techniques consider only the time interval between `KU` and `KD` events, i.e., `KU-KU` time, while more recent studies [4, 28] demonstrate the importance of adding additional features, such as `KD-KD` time. Similar to [28], UNAGI associates features to all the possible time intervals defined over `KD` and `KU` events, that is `KD-KU` time, `KU-KD` time, `KD-KD` time, and `KU-KU` time.

Detection

The output of the feature extraction phase is a vector containing all the features considered: keystroke timings and n-graphs-associated sensor statistical metrics. Common machine learning practices dictate normalizing such a vector so that the value ranges for all its elements are comparable [57]. Normalization ensures that the maximum and minimum values for each element are constant across all the vectors and all other values are linearly distributed. Such labeled feature vectors are suitable for standard supervised machine learning algorithms [57].

In detail, our problem can be addressed by standard threshold-based binary classification algorithms, a comparison of which can be found in [29]. The current UNAGI implementation supports one-class SVM, Naive Bayes, k-nearest neighbors (kNN), and the "mean algorithm". The latter is similar to kNN, but compares the test samples against the mean training sample—instead of all the training samples. Similar to [29], UNAGI considers the following distance metrics: Euclidean, Euclidean normed, Manhattan, Manhattan scaled, Mahalanobis. We also experimented with our own weighted metrics, where the weights represent the "importance" of a given feature in the vector:

- ***Euclidean Weighted***: $\mathrm{ew}(p,q) = \sqrt{\frac{\sum_{i=1}^{n} w_i^2 (p_i - q_i)^2}{\sum_{i=1}^{n} w_i^2}}$.
- ***Euclidean Normed Weighted***: $\mathrm{enw}(p,q) = \frac{ew(p,q)}{\|p\|_2 \|q\|_2}$.
- ***Manhattan Weighted***: $\mathrm{mw}(p,q) = \frac{\sum_{i=1}^{n} w_i |p_i - q_i|}{\sum_{i=1}^{n} w_i}$.
- ***Manhattan Scaled Weighted***: $\mathrm{msw}(p,q) = \frac{\sum_{i=1}^{n} \frac{w_i |p_i - q_i|}{a_i}}{\sum_{i=1}^{n} w_i}$.

For two vectors p and q and a vector of weights w, we denote its elements by p_i, q_i and w_i ($1 \leq i \leq n$, where n is the size of the vectors). Vector a represents the mean absolute deviation of each feature in the training vectors, while $\|v\|_2$ denotes the second norm of the vector v.

Since our preliminary tests revealed poor accuracy for SVM, Naive Bayes, and Mahalanobis distance-based algorithms, we decided to ignore such algorithms in further experiments. Our analysis also showed that $k = 1$ is the optimal parameter for kNN, a configuration which we adopted throughout all our experiments.

Testing

To test our classifiers, we use the leave-one-out cross-validation—an instance of k-fold cross-validation with k set to the number of samples for a specific user. This testing strategy performs particularly well when the training data are small [57], a scenario which reflects our dataset of approximately 40 samples per

user. In the testing phase, we evaluate the accuracy for each user separately and aggregate the results only at the end of the process. Classification thresholds are chosen separately for each user based on the training data, a strategy which drastically improves the final accuracy. For each user, we perform the following steps. The training data for one user is derived from the set of all his samples except for a predetermined sample z. The testing data are derived from the set containing the sample z. Samples from all other users are considered impostors samples. Accuracy is computed for each user and all the possible values of z.

On average, each classifier is tested on 370 valid user samples and 130,000 impostor user samples, while trained using only valid user training samples.

5 Evaluation

In this section, we report on the experimental evaluation of our solution, starting with the description of the experimental setup and the error metrics considered.

Experimental Setup

For our experiments, we gathered samples from a number of test subjects typing predetermined passwords. To directly compare our results with prior work in the area—which generally evaluated accuracy in a similar controlled setting—we conducted our experiments with the subjects seated typing on a mobile device, allowing all the interested students in our department (20) to participate in the experiment and negotiate the number of password repetitions (40) in advance. For our experiments, we used a Samsung Nexus S with a soft keyboard in landscape mode, resulting in a 17Hz sensor sampling frequency for each axis.

We evaluated UNAGI with two passwords, i.e., `internet` and `satellite`, negotiated in number, length, and type in advance with the test subjects. This strategy was sought to obtain the best usability-accuracy tradeoff possible and prevent measurement bias. During the experiments, we allowed each typing error to invalidate the current sample and request the subject to produce a new sample.

We evaluated our techniques in three different configurations: keystroke timings only, sensor data only, and combination thereof. For our sensor data analysis, we considered different n-graphs: *1*-, *1.5*-, *2*-, *2.5*-, *3*-, *3.5*-, *4*-, and *4.5*-n-graphs. For each choice of n, we considered all the possible combinations with step $S = 0.5$ (i.e., a distinct n-graph starting at every 0.5 step). For our keystroke timing analysis, we first considered all the possible combinations of KD and KU events—*0.5*-graphs and *1*-graphs with step $S = 0.5$. To compare sensor data and keystroke timing results, we also evaluated longer n-graphs (*1.5*-, *2*-, *2.5*-, *3*-, *3.5*-, *4*-, and *4.5*-n-graphs). To compute our weighted distances, we relied on the weights derived by SVM feature ranking based on the training data.

In order to compare different authentication systems, we need a consistent way to measure accuracy. Two standard error metrics used in the literature [28] are FAR (*false acceptance rate*), which indicates the fraction of impostor access

attempts identified as valid users, and FRR (*false rejection rate*), which indicates the fraction of valid user attempts identified as impostors. FAR and FRR are strictly correlated and can be controlled by a threshold, which establishes the conservativeness of the approach and affects FAR and FRR in opposite ways. To obtain a single value summarizing the accuracy of a system, prior approaches described in the literature [28] typically relied on the EER (equal error rate), which is defined as the value of FAR (or FRR) when FAR and FRR are identical (with the threshold tuned accordingly). We considered only $EERs$ to measure the accuracy of our techniques in our evaluation.

Accuracy

Figure 5 depicts the accuracy of our techniques for different n-graph sizes, considering only keystroke timings (and no sensor data) and the minimum EER found across all our detection algorithms. From the figure, we can observe that increasing the n-graph size has a negative impact on the accuracy. This behavior confirms the importance of using a fine-grained feature characterization strategy for keystroke timings. In addition, we obtained the most accurate results when using only *0.5*-graphs (`KU-KD` time and `KD-KU` time), a result which contradicts some of the analyses reported in prior studies in the area [4]. This suggests that traditional feature selection strategies for keystroke dynamics may have to be carefully redesigned for touch screen devices. In addition, results for the `internet` password revealed slightly better results. This suggests that the choice of the password may affect the final accuracy in nontrivial ways. Further investigation is necessary to predict the quality of a particular password for keystroke or sensor dynamics purposes.

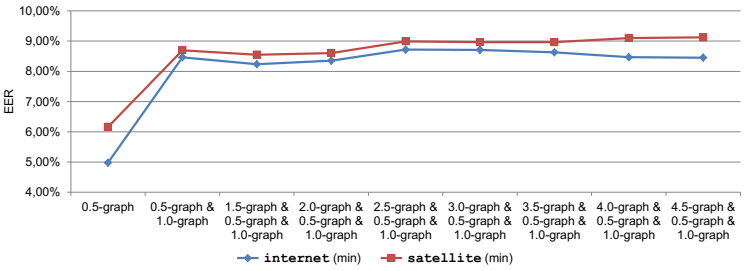

Fig. 5. Accuracy (EER) for varying n-graph sizes (keystroke timings only)

Figure 6 depicts the accuracy of our techniques for different n-graph sizes, considering only sensor data and the minimum EER found across all our detection algorithms. As shown in the figure, the accuracy improves—although at a slow pace—with the n-graph size. This behavior demonstrates that, in contrast to keystroke timings, a coarser-grained feature characterization strategy is more

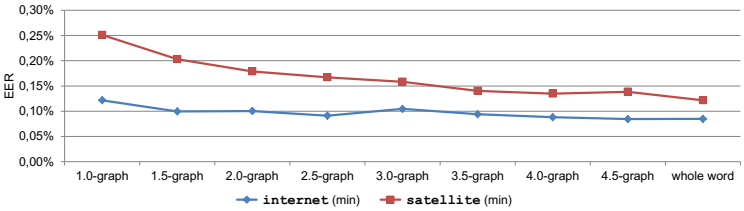

Fig. 6. Accuracy (EER) for varying n-graph sizes (sensor data only)

effective for sensor data. We believe this result stems from statistical analysis providing more stable and accurate results on a larger amount of data.

Figure 7 depicts the accuracy of our techniques for the different detection algorithms considered. As shown in the figure, we found "kNN ($k = 1$) Manhattan weighted" and "kNN ($k = 1$) Manhattan scaled weighted" to be the best performing algorithms, with the former resulting in the lowest (0.08%) EER using only sensor data. In addition, the figure shows that algorithms based on weighted distances outperformed unweighted ones in almost all cases.

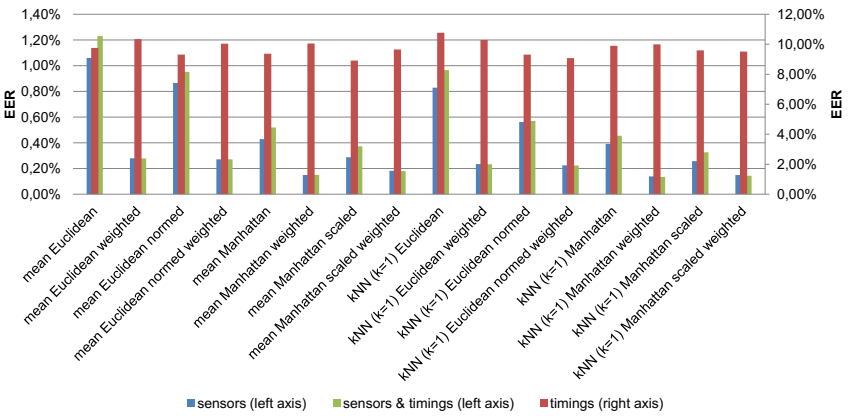

Fig. 7. Accuracy (EER) for the different detection algorithms considered

Another concern we wish to address is how the sensor sampling frequency impacts the accuracy of our authentication techniques. To this end, we repeated our experiments for different values of the sampling frequency. The results are reported in Figure 8. As shown in the figure, decreasing the sampling frequency even by a factor of 2 does not significantly lower the accuracy. Reasonably low frequencies are instead sufficient to achieve accurate results. This is encouraging and suggests that sensor-enhanced keystroke dynamics could provide high accuracy even for low-end devices. In addition, in fixed-text analyses, sensors are

used only for short time intervals, with minimal impact on battery usage. Finally, the trend depicted in the figure seems to suggest that increasing the sampling frequency further (i.e., higher than 17Hz) does not lead to significant accuracy benefits. More sophisticated sensor-based devices, however, may provide more accurate results. Note that the empirical evidence presented here is based on statistical analysis and should not be regarded as conclusive in the general case.

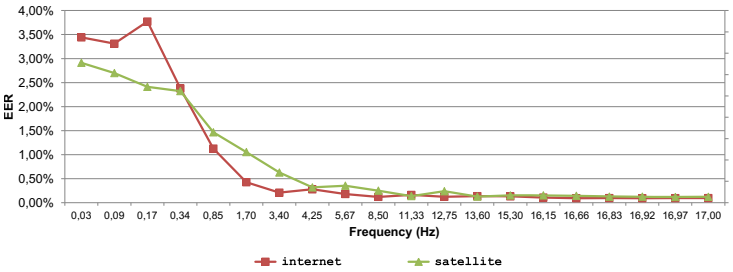

Fig. 8. Accuracy (EER) for varying sensor sampling frequencies (sensor data only)

Finally, Table 1 reports the most relevant sensor-related features according to the weights computed by SVM feature ranking for our weighted distance metrics. The weights are averaged over the two passwords and obtained using whole-word analysis and only sensor data. Our results show that the Z axis is less relevant than the other axes and that the accelerometer is much more relevant than the gyroscope. Interestingly, this also suggests that sensor-enhanced keystroke dynamics requires different feature selection strategies than prior machine learning techniques that relied on sensor data to perform side channel attacks [10, 40].

Table 1. Top 10 features for movement sensors (mean SVM weights)

Mean Weight	Feature	Sensor	Axis
128	Average value	Accelerometer	Y
91	Average value	Accelerometer	X
78	Root mean square	Accelerometer	X
61	Average value	Accelerometer	Z
38	Sum of positive values	Accelerometer	Y
19	Sum of positive values	Accelerometer	Z
16	Sum of negative values	Accelerometer	X
11	Root mean square	Accelerometer	Y
11	Root mean square	Gyroscope	X
11	Standard deviation	Gyroscope	X

To summarize, across all the configurations, our best detector and password achieved 4.97% EER using only keystroke timings and 0.08% EER using only sensor data. Our results also show that combining sensor data and keystroke timings does not substantially improve the accuracy when compared to using only

sensor data, with only marginal (e.g., ±0.01%) variations for our best-performing detectors—although it may improve robustness against human or synthetic attacks, but further investigation is necessary to draw general conclusions.

Table 2 compares our accuracy results with prior keystroke dynamics techniques. As shown in the table, accurate comparisons are not always possible, given that some studies report only FAR/FRR and other studies rely on non-standard experimental settings that may overestimate the final accuracy reported (see "Notes" column for details). Encouragingly, prior results obtained on mobile devices with software keyboards are comparable to ours (4.97% EER with only keystroke timings), which confirms the soundness of our experimental analysis. Unfortunately, we cannot directly compare our sensor-related accuracy results with prior work, given that we are the first to explore sensor-enhanced keystroke dynamics on mobile devices. Recent work by Tasi et al. [55] comes conceptually close, investigating how to improve the accuracy of keystroke dynamics techniques using pressure information. Their reported EER values, however, are as high as 8.4%, with pressure information only introducing relatively small accuracy improvements with respect to their keystroke timing-only configuration (11.4% EER). In contrast, our experience with Unagi demonstrates that a carefully designed feature extraction strategy based on sensor-sampled distributions can drastically improve keystroke dynamics accuracy (i.e., from 4.97% EER to 0.08% EER, with our best detector and password).

Table 2. Accuracy comparison with prior keystroke dynamics techniques

Keyboard	Source	Accuracy	Notes
Hardware (PC)	[25]	13.30% FRR, 0.17% FAR	
	[34]	1.10% FRR, 0.00% FAR	Small dataset.
	[42]	0.00% EER	Small dataset.
	[7]	4.00% FRR, 0.01% FAR	Long password (683 characters).
	[4]	1.45% FRR, 1.89% FAR	Allows 1 authentication failure.
	[26]	3.80% EER	
	[28]	7.10% EER	
Hardware (Mobile device)	[13]	10.40% EER	
	[27]	12.20% EER	
	[12]	13.59% EER	
	[24]	4.00% EER	Use of artificial rhythms.
	[60]	0.00% FRR, 2.00% FAR	Allows 1 authentication failure.
Software (Mobile device)	[23]	7.50% EER	Allows 1 authentication failure.
	[56]	5.26% FRR, 8.31% FAR	
	[55]	8.40% EER	

6 Related Work

In the following, we survey the most relevant techniques in the area and refer the interested reader to more complete surveys [17].

Keystroke Dynamics on Hardware Keyboards

Pioneering work in the area of keystroke dynamics was undertaken by Gaines et al. in 1980 [20]. Seven secretaries typed a predetermined text and their actions analyzed using statistical analysis. The authors concluded that, using mainly digraph latencies, users can be distinguished according to their typing behavior. Further experiments conducted by Leggett et al. [32] confirmed the original intuitions in [20]. Joyce et al. [25] presented the first analytical keystroke dynamics accuracy evaluation, reporting a 13.3% FRR and 0.17% FAR. De Ru et al. [46] first proposed fuzzy classification algorithms, later also adopted by other researchers. In 1997, Monrose and Rubin [41] suggested using keystroke dynamics as a free-text authentication mechanism (amenable to continuous authentication) resulting in 90% accuracy in identifying users. The same authors reported a 92.14% accuracy for fixed-text analysis three years later. Around that time, Lin [34] reported much higher accuracy results (i.e., 1.1% FRR and 0% FAR) using neural networks, although he considered only one sample per user, likely overestimating the real accuracy. Similarly, Obaidat and Sadoun [42] reported high accuracy results using neural networks (0% FRR and 0% FAR), but considered a very small number of impostor samples. Bergadano et al. [7], in contrast, proposed using distance-based classification algorithms and reported 4% FRR and 0.01% FAR. Such results, however, were obtained using a large fixed text length (683 characters). Araujo et al. [4] first proposed combining `KD-KU` times and `KU-KD` times with `KD-KD` times, reporting 1.45% FRR and 1.89% FAR, but only when raising an alarm after two consecutive failed authentication attempts. Kotani and Horii [31] built their own keyboard-equipped device to be able to measure finger pressure while typing. The authors reported a 2.4% EER (keystroke timings only) using statistical analysis with fuzzy logic and neural networks. In [26], Kang et al. suggested periodic retraining to mitigate the impact of variations in typing patterns over time. They considered a "sliding window" approach, where a fixed number of recent patterns were used to train a classifier, ultimately reporting a 3.8% EER with their best detection algorithm. In another direction, Killourhy and Maxion [28] analyzed the factors influencing keystroke dynamics error rates. Using a 10-character password and statistical analysis, they concluded that the detection algorithm, the amount of training, and the ability to update training data have the strongest impact on the final detection accuracy. They also found other factors such as impostor practice and variations in the feature set to be much less relevant for the final accuracy. Their analysis reported an accuracy of 7.1% EER for their best-performing detector—i.e., Manhattan (scaled) algorithm. In their earlier work [29], the same authors experimented with 51 subjects and 14 algorithms. Their earlier analysis reported an accuracy of 9.6% EER for the same (best-performing) detector.

Keystroke Dynamics on Hardware Keyboards for Mobile Devices

One of the first keystroke dynamics techniques for mobile devices was proposed by Clarke et al. [14] on a Nokia 5510 device with a numeric keyboard. Using

neural networks, the authors reported a 11.3% EER for 4-digit password, 10.4% for 9-digit password, and 24.5% for free text. Karatzouni and Clarke [27] reported comparable results on similar devices (12.2% EER). Campisi et al. [12] analyzed a typing scenario with alphabetic strings on numeric keyboards and obtained a 13.59% EER using a statistical classifier. Hwang et al. [24] reported accuracy improvements for short PIN lengths when using artificial rhythms and tempo cues. This strategy decreased their $EERs$ from 13% to 4%. Zahid et al. [60] developed a tri-mode continuous verification system. Using a fuzzy classifier and particle swarm optimizations, they obtained a 0% FRR and 2% FAR, but only when using multiple verification systems.

Keystroke Dynamics on Software Keyboards for Mobile Devices

Saevanee and Bhattarakosol first evaluated the impact of finger pressure on keystroke dynamics techniques for mobile devices [47], but only performed simulated experiments using a notebook touchpad. They reported a 1% EER using a kNN algorithm and later obtained similar results using neural networks [48]. More recent studies on real mobile devices seem to suggest that pressure has a much smaller accuracy impact in practice, ultimately resulting in a 8.4% EER when combined with keystroke timings [55]. Huang et al. [23] first explored traditional keystroke dynamics techniques on software keyboards for mobile devices and reported a 7.5% EER, but only when raising an alarm after 2 consecutive failed authentication attempts. Trojahn and Ortmeier [56] extended the analysis to both numeric and alphabetic passwords and both numeric and QWERTY keyboards, reporting nontrivial variations across configurations, with $FRRs$ and $FARs$ in the range of 5.26%-8.75% and 8.31%-12.13%, respectively.

Sensor-Based Side Channel Attacks

A number of studies have recently demonstrated the feasibility of side channel attacks on mobile devices using movement sensor data. Typical attacks exploit the intuition that statistical analysis of sensor data provides a strong characterization of a given user, an idea which we used as a foundation for sensor-enhanced keystroke dynamics. Cai and Chen [10] presented a 70%-accuracy keylogging attack on numeric touchscreen keyboards which relies solely on sensor data. In contrast to our results, they observed that data read from the gyroscope is more user independent than data read from the accelerometer. Miluzzo et al. [40] relied on gyroscope and accelerometer data to infer the icon activated by the user in iOS and reported a 90% accuracy. Owusu et al. [44], in contrast, relied only on accelerometer data to infer complete sequences of characters. The authors reported an average of 4.5 attempts to guess a 6-characters passwords. Their probabilistic model based on statistical analysis is similar, in spirit, to our feature extraction strategy for sensor data. Xu et al. [59] proposed *TapLogger*, an accelerometer-based keylogger for numeric soft keyboards. The authors reported a 97.5% accuracy for 8-digit passwords and 3 authentication attempts. Aviv et al. [6] relied on accelerometer data and keystroke timings to infer 4-digit

PINs and unlock screen patterns. The authors reported an accuracy of 43% and 73% for the two scenarios considered (respectively), using 5 authentication attempts in a controlled setting. Souya Faria and Kim [52] presented an attack based on the analysis of mechanical vibrations inferred by accelerometer data. The authors reported key recognition rates of 98.4% on an ATM keypad, 76.7% on a PIN pad on a hard surface and 82.1% on a PIN pad held with one hand.

Gesture-Based Authentication

Guerra Casanova et al. [21] first proposed an authentication technique based on user gestures for mobile devices. Their approach relied on accelerometer data and reported a 2.5% EER. Similarly, Kolly et al. [30] proposed touch events to authenticate users interacting with a mobile device. The authors reported 80% accuracy using a Naive Bayes classification algorithm based only on a few touch events. Han et al. [22] suggested using accelerometer data to infer the GPS coordinates of a mobile device within a 200m radius from the real location. Frank et al. [19] presented a continuous authentication system based on 30 touch-based gestures. Their SVM and kNN detection algorithms resulted in 0%-4% EER depending on whether training and testing were performed during the same user session. Liu [35] presented a detailed study on mobile device sensors and discussed novel applications enabled by sensor data. Meng et al. [39] proposed a post-login continuous authentication system with 0.13% FRR and 4.66% FAR. To obtain the reported accuracy, they relied on a special glove equipped with accelerometers and interacting with a touch screen using particular gestures. Damopoulos et al. [17] proposed a continuous authentication system using only touchscreen gestures. The authors reported a low 1% EER using predetermined touch patterns. Recent proposals described in [18, 33, 51] have often reported even lower $EERs$ in particular scenarios, as low as 0.5% EER, in particular, when using a fine-grained stroke characterization strategy [51]. Gesture-based authentication schemes, however, have been already shown extremely vulnerable to simple statistical attacks, which can easily yield substantial EER increases while relying only on general population statistics [50].

7 Conclusion

In this paper, we presented *sensor-enhanced keystroke dynamics*, a new biometric authentication mechanism for mobile devices. The key intuition is to leverage movement sensor data to strengthen the user characterization guarantees provided by traditional keystroke dynamics techniques, an idea inspired by emerging side channel attacks on sensor-equipped mobile devices [6, 10, 11, 40, 44, 52, 59].

To demonstrate the effectiveness of our approach, we implemented Unagi, an Android prototype based on the proposed sensor-enhanced keystroke dynamics mechanism. Unagi relies on sensor data (i.e., accelerometer and gyroscope) and keystroke timings to implement a general-purpose fixed-text authentication system. Unagi outperforms prior biometric techniques for mobile devices in terms

of both accuracy and robustness against attacks. In particular, we demonstrated how a careful feature extraction strategy coupled with standard machine learning techniques can produce a high-accuracy detector, even for relatively low sensor sampling frequencies and short passwords. Our results confirm that movement sensor provides extremely accurate information to characterize user behavior and identify unique biometric features suitable for authentication purposes.

In addition, and somewhat surprisingly, our results demonstrate that the accuracy yielded by sensor-based features outperforms the accuracy of standard keystroke dynamics features (i.e., keystroke timings) by up to two orders of magnitude (i.e., 0.08% EER vs. 4.97% EER with our best detector/password, respectively) and that their combination provides little accuracy benefits compared to a sensor-only configuration. With a EER of only 0.08% reported by the best detector/password in our experiments, we believe ours is the first promising attempt to fill the gap between traditional keystroke dynamics techniques and the accuracy required in real-world authentication systems.

We are currently considering three main directions for future work. First, we are planning to investigate techniques to further increase the accuracy of sensor-enhanced keystroke dynamics (e.g., by using more sophisticated sensors or detection algorithms). The gold standard is to reach a FRR of less than 1%, with a FAR of no more than 0.001%—as specified by the European standard for access-control systems (EN-50133-1) [29]. Second, we are planning to investigate techniques to maximize the accuracy of sensor-enhanced keystroke dynamics in both uncontrolled and free-text authentication scenarios, for instance by employing noise-suppression techniques to improve the quality of the sensor-sampled distributions. Finally, we are planning to thoroughly evaluate the robustness of sensor-enhanced keystroke dynamics against human and synthetic attacks [50].

Acknowledgements. We would like to thank the anonymous reviewers for their insightful comments. Cristiano Giuffrida is supported by the Re-Cover project funded by NWO. Mauro Conti is supported by a Marie Curie Fellowship funded by the European Commission (grant PCIG11-GA-2012-321980) and by a PRIN project funded by the Italian MIUR (grant 20103P34XC).

References

1. Cisco visual networking index: Global mobile data traffic forecast update (2012 -2017), http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html
2. The Symantec smartphone honey stick project, http://www.symantec.com/content/en/us/about/presskits/b-symantec-smartphone-honey-stick-project.en-us.pdf
3. With 1.6 million smart phones stolen last year, efforts under way to stem the losses, http://www.consumerreports.org/cro/news/2013/06/with-1-6-million-smart-phones-stolen-last-year-efforts-under-way-to-stem-the-losses/index.htm

4. Araujo, L., Sucupira Jr., L.H.R., Lizarraga, M., Ling, L., Yabu-Uti, J.B.T.: User authentication through typing biometrics features. IEEE Trans. Signal Process. 53(2), 851–855 (2005)
5. Aviv, A.J., Gibson, K., Mossop, E., Blaze, M., Smith, J.M.: Smudge attacks on smartphone touch screens. In: Proc. of the 4th USENIX Conf. on Offensive Technologies, pp. 1–7 (2010)
6. Aviv, A.J., Sapp, B., Blaze, M., Smith, J.M.: Practicality of accelerometer side channels on smartphones. In: Proc. of the 28th Annual Computer Security Appl. Conf., pp. 41–50 (2012)
7. Bergadano, F., Gunetti, D., Picardi, C.: User authentication through keystroke dynamics. ACM Trans. Inf. Syst. Secur. 5(4), 367–397 (2002)
8. Brown, P.F., de Souza, P.V., Mercer, R.L., Pietra, V.J.D., Lai, J.C.: Class-based n-gram models of natural language. Comput. Linguist. 18(4), 467–479 (1992)
9. Burnett, M.: 10,000 top passwords, `http://xato.net/passwords/more-top-worst-passwords/`
10. Cai, L., Chen, H.: TouchLogger: Inferring keystrokes on touch screen from smartphone motion. In: Proc. of the Sixth USENIX Workshop on Hot Topics in Security, p. 9 (2011)
11. Cai, L., Chen, H.: On the practicality of motion based keystroke inference attack. In: Katzenbeisser, S., Weippl, E., Camp, L.J., Volkamer, M., Reiter, M., Zhang, X. (eds.) Trust 2012. LNCS, vol. 7344, pp. 273–290. Springer, Heidelberg (2012)
12. Campisi, P., Maiorana, E., Lo Bosco, M., Neri, A.: User authentication using keystroke dynamics for cellular phones. IET Signal Processing 3(4), 333–341 (2009)
13. Clarke, N.L., Furnell, S.M.: Authenticating mobile phone users using keystroke analysis. Int'l J. Inf. Secur. 6(1), 1–14 (2006)
14. Clarke, N.L., Furnell, S.M., Lines, B.M., Reynolds, P.L.: Keystroke dynamics on a mobile handset: A feasibility study. Information Management & Computer Security 11(4), 161–166 (2003)
15. Clarke, N.L., Furnell, S.M.: Authentication of users on mobile telephones-A survey of attitudes and practices. Computers & Security 24(7), 519–527 (2005)
16. Conti, M., Zachia-Zlatea, I., Crispo, B.: Mind how you answer me!: Transparently authenticating the user of a smartphone when answering or placing a call. In: Proc. of the Sixth ACM Symp. on Information, Computer and Communications Security, pp. 249–259 (2011)
17. Damopoulos, D., Kambourakis, G., Gritzalis, S.: From keyloggers to touchloggers: Take the rough with the smooth. Computers & Security 32, 102–114 (2013)
18. De Luca, A., Hang, A., Brudy, F., Lindner, C., Hussmann, H.: Touch me once and i know it's you!: Implicit authentication based on touch screen patterns. In: Proc. of the SIGCHI Conf. on Human Factors in Computing Systems, pp. 987–996 (2012)
19. Frank, M., Biedert, R., Ma, E., Martinovic, I., Song, D.: Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Trans. Inf. Forensics and Security 8(1), 136–148 (2013)
20. Gaines, R.S., Lisowski, W., Press, S.J., Shapiro, N.: Authentication by keystroke timing. Tech. rep. (1980)
21. Guerra Casanova, J., Avila, C., de Santos Sierra, A., Bailador del Pozo, G., Jara Vera, V.: Acceleration axis selection in biometric technique based on gesture recognition. In: Proc. of the Sixth Int'l Conf. on Intelligent Information Hiding and Multimedia Signal Processing, pp. 360–363 (2010)
22. Han, J., Owusu, E., Nguyen, L., Perrig, A., Zhang, J.: ACComplice: Location inference using accelerometers on smartphones. In: Proc. of the Fourth Int'l Conf. on Communication Systems and Networks, pp. 1–9 (2012)

23. Huang, X., Lund, G., Sapeluk, A.: Development of a typing behaviour recognition mechanism on android. In: Proc. of the 11th Int'l Conf. on Trust, Security and Privacy in Computing and Communications, pp. 1342–1347 (2012)
24. Hwang, S.S., Cho, S., Park, S.: Keystroke dynamics-based authentication for mobile devices. Computers & Security 28(1-2), 85–93 (2009)
25. Joyce, R., Gupta, G.: Identity authentication based on keystroke latencies. Communications of The ACM 33(2), 168–176 (1990)
26. Kang, P., Hwang, S.-s., Cho, S.: Continual retraining of keystroke dynamics based authenticator. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 1203–1211. Springer, Heidelberg (2007)
27. Karatzouni, S., Clarke, N.: Keystroke analysis for thumb-based keyboards on mobile devices. In: Venter, H., Eloff, M., Labuschagne, L., Eloff, J., Solms, R. (eds.) Proc. of the 22nd IFIP Int'l Information Security Conf., pp. 253–263 (2007)
28. Killourhy, K., Maxion, R.: Why did my detector do *that*?!: Predicting keystroke-dynamics error rates. In: Jha, S., Sommer, R., Kreibich, C. (eds.) RAID 2010. LNCS, vol. 6307, pp. 256–276. Springer, Heidelberg (2010)
29. Killourhy, K.S., Maxion, R.A.: Comparing anomaly-detection algorithms for keystroke dynamics. In: Proc. of the Int'l Conf. on Dependable Systems and Networks, pp. 125–134 (2009)
30. Kolly, S.M., Wattenhofer, R., Welten, S.: A personal touch: Recognizing users based on touch screen behavior. In: Proc. of the Third Int'l Workshop on Sensing Applications on Mobile Phones, pp. 1–5 (2012)
31. Kotani, K., Horii, K.: Evaluation on a keystroke authentication system by keying force incorporated with temporal characteristics of keystroke dynamics. Behaviour & Information Technology 24(4), 289–302 (2005)
32. Leggett, J., Williams, G.: Verifying identity via keystroke characteristics. Int'l J. Man-Mach. Stud. 28(1), 67–76 (1988)
33. Li, L., Zhao, X., Xue, G.: Unobservable re-authentication for smartphones. In: Proc. of the 20th Network and Distributed System Security Symp. (2013)
34. Lin, D.T.: Computer-access authentication with neural network based keystroke identity verification. In: Proc. of the Int'l Conf. on Neural Networks, pp. 174–178 (1997)
35. Liu, M.: A study of mobile sensing using smartphones. Int'l J. of Distributed Sensor Networks 2013(2013)
36. Maiorana, E., Campisi, P., González-Carballo, N., Neri, A.: Keystroke dynamics authentication for mobile phones. In: Proc. of the ACM Symp. on Applied Computing, pp. 21–26 (2011)
37. Mantyjarvi, J., Lindholm, M., Vildjiounaite, E., Makela, S.M., Ailisto, H.: Identifying users of portable devices from gait pattern with accelerometers. In: Proc. of the Int'l Conf. on Acoustics, Speech, and Signal Processing, pp. 973–976 (2005)
38. Meng, T.C., Gupta, P., Gao, D.: I can be you: Questioning the use of keystroke dynamics as biometrics. In: Proc. of the 20th Network and Distributed System Security Symp. (2013)
39. Meng, Y., Wong, D.S., Schlegel, R., Kwok, L.-F.: Touch gestures based biometric authentication scheme for touchscreen mobile phones. In: Kutyłowski, M., Yung, M. (eds.) Inscrypt 2012. LNCS, vol. 7763, pp. 331–350. Springer, Heidelberg (2013)
40. Miluzzo, E., Varshavsky, A., Balakrishnan, S., Choudhury, R.R.: Tapprints: Your finger taps have fingerprints. In: Proc. of the 10th Int'l Conf. on Mobile Systems, Applications, and Services, pp. 323–336 (2012)
41. Monrose, F., Rubin, A.: Authentication via keystroke dynamics. In: Proc. of the Fourth ACM Conf. on Computer and Communications Security, pp. 48–56 (1997)

42. Obaidat, M., Sadoun, B.: Verification of computer users using keystroke dynamics. IEEE Trans. Syst. Man, Cybern. B, Cybern. 27(2), 261–269 (1997)
43. Okumura, F., Kubota, A., Hatori, Y., Matsuo, K., Hashimoto, M., Koike, A.: A study on biometric authentication based on arm sweep action with acceleration sensor. In: Proc. of the Int'l Symp. on Intelligent Signal Processing and Communications, pp. 219–222 (2006)
44. Owusu, E., Han, J., Das, S., Perrig, A., Zhang, J.: Accessory: Password inference using accelerometers on smartphones. In: Proc. of the 12th Workshop on Mobile Computing Systems and Applications, pp. 1–6 (2012)
45. Rahman, K., Balagani, K., Phoha, V.: Snoop-forge-replay attacks on continuous verification with keystrokes. IEEE Trans. on Information Forensics and Security 8(3), 528–541 (2013)
46. de Ru, W.G., Eloff, J.H.P.: Enhanced password authentication through fuzzy logic. IEEE Expert 12(6), 38–45 (1997)
47. Saevanee, H., Bhatarakosol, P.: User authentication using combination of behavioral biometrics over the touchpad acting like touch screen of mobile device. In: Proc. of the Int'l Conf. on Computer and Electrical Engineering, pp. 82–86 (2008)
48. Saevanee, H., Bhattarakosol, P.: Authenticating user using keystroke dynamics and finger pressure. In: Proc. of the Sixth IEEE Conf. on Consumer Communications and Networking, pp. 1078–1079 (2009)
49. Serwadda, A., Phoha, V.V.: Examining a large keystroke biometrics dataset for statistical-attack openings. ACM Trans. Inf. Syst. Secur. 16(2), 1–30 (2013)
50. Serwadda, A., Phoha, V.V.: When kids' toys breach mobile phone security. In: Proc. of the 2013 ACM Conf. on Computer and Communications Security, pp. 599–610 (2013)
51. Shahzad, M., Liu, A.X., Samuel, A.: Secure unlocking of mobile touch screen devices by simple gestures: You can see it but you can not do it. In: Proc. of the 19th Annual Int'l Conf. on Mobile Computing and Networking, pp. 39–50 (2013)
52. de Souza Faria, G., Kim, H.Y.: Identification of pressed keys from mechanical vibrations. IEEE Trans. Inf. Forensics and Security 8(7), 1221–1229 (2013)
53. Stefan, D., Shu, X., Yao, D.: Robustness of keystroke-dynamics based biometrics against synthetic forgeries. Computers & Security 31(1), 109–121 (2012)
54. Tari, F., Ozok, A.A., Holden, S.H.: A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords. In: Proc. of the Second Symp. on Usable Privacy and Security, pp. 56–66 (2006)
55. Tasi, C.J., Chang, T.Y., Cheng, P.C., Lin, J.H.: Two novel biometric features in keystroke dynamics authentication systems for touch screen devices. Security and Communication Networks (2013)
56. Trojahn, M., Ortmeier, F.: Biometric authentication through a virtual keyboard for smartphones. Int'l J. Computer Science & Information Technology 4(5) (2012)
57. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical Machine Learning Tools and Techniques (2011)
58. Xu, Y., Heinly, J., White, A.M., Monrose, F., Frahm, J.M.: Seeing double: Reconstructing obscured typed input from repeated compromising reflections. In: Proc. of the 2013 ACM Conf. on Computer and Communications Security, pp. 1063–1074 (2013)
59. Xu, Z., Bai, K., Zhu, S.: TapLogger: Inferring user inputs on smartphone touchscreens using on-board motion sensors. In: Proc. of the Fifth ACM Conf. on Security and Privacy in Wireless and Mobile Networks, pp. 113–124 (2012)
60. Zahid, S., Shahzad, M., Khayam, S.A., Farooq, M.: Keystroke-based user identificationon smart phones. In: Kirda, E., Jha, S., Balzarotti, D. (eds.) RAID 2009. LNCS, vol. 5758, pp. 224–243. Springer, Heidelberg (2009)