DialerAuth: A Motion-assisted Touch-based Smartphone User Authentication Scheme

Attaullah Buriro
University of Trento
Trento, Italy
attaullah.buriro@unitn.it

Bruno Crispo
University of Trento
Trento, Italy
bruno.crispo@unitn.it

Sandeep Gupta
University of Trento
Trento, Italy
sandeep.gupta@unitn.it

Filippo Del Frari
University of Trento
Trento, Italy
filippo.delfrari@unitn.it

ABSTRACT

This paper introduces DialerAuth - a mechanism which leverages the way a smartphone user taps/enters any "*text-independent*" 10-digit number (replicating the dialing process) and the hand's micro-movements she makes while doing so. DialerAuth authenticates the user on the basis of timing differences in the entered 10-digit strokes. DialerAuth provides enhanced security by leveraging the transparent and unobservable layer based on another modality - user's hand micro-movements. Furthermore, DialerAuth increases the usability and acceptability by utilizing the users' familiarity with the dialing process and the flexibility of choosing any combination of 10-digit number. We implemented DialerAuth for both data collection and proof-of-concept real-time analysis. We collected, in total ≈10500 legitimate samples involving 97 users, through an extensive unsupervised field experiment, to evaluate the effectiveness of DialerAuth. Analysis using one-class Multilayer Perceptron (MLP) shows a TAR of 85.77% in identifying the genuine users. Security analysis involving ≈240 adversarial attempts proved DialerAuth as significantly resilient against random and mimic attacks. A usability study based on System Usability Scale (SUS) reflects a positive feedback on user acceptance (SUS score = 73.29).

CCS CONCEPTS

• **Security and privacy → Authentication**; **Biometrics**; • **Human-centered computing → Mobile devices**;

KEYWORDS

Smartphone Authentication, Behavioral biometrics, Sensors

ACM Reference format:
Attaullah Buriro, Bruno Crispo, Sandeep Gupta, and Filippo Del Frari. 2018. DialerAuth: A Motion-assisted Touch-based Smartphone User Authentication Scheme. In *Proceedings of Eighth ACM Conference on Data and Application Security and Privacy, Tempe, AZ, USA, March 19–21, 2018 (CODASPY'18),* 10 pages.
https://doi.org/10.1145/3176258.3176318

CODASPY'18, March 19–21, 2018, Tempe, AZ, USA

ACM ISBN 978-1-4503-5632-9/18/03…$15.00
https://doi.org/10.1145/3176258.3176318

1 INTRODUCTION

Smartphone user authentication is the process of verifying the identity of the claimant and to ensure that only the authenticated and authorized user is granted access to the smartphone. Popular smartphone user authentication schemes, based on "something the user knows" (e.g., PINs/passwords, graphical patterns), have shown to have usability issue (because they cause inconvenience) [15] and also security issues (due to their vulnerability to different attacks) [24]. A recent study reported that users considered login with PINs and passwords more annoying than any other technology related problems, such as lack of coverage, small screen size or low voice quality [18].

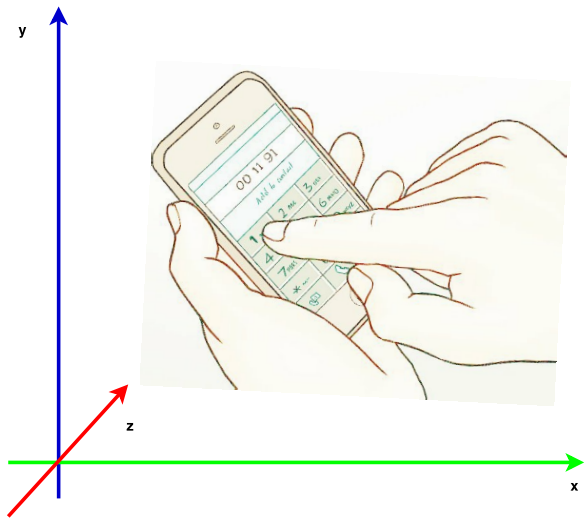


Figure 1: Dialing process in 3d space.

The recent introduction of physiological biometrics based on "something the user is", i.e., facial, fingerprint and iris recognition,

have somehow mitigated the problems with the user input, however, some usability [10] and security [1] issues still remain unsolved.

This led researchers to investigate the feasibility of methods based on something the user does - often termed as behavioral biometrics. Behavioral biometrics offer various advantages over physiological counterparts. One of the main advantages is that the behavioral patterns can be used transparently. More importantly, data collection does not require any special dedicated hardware. Consequently, researchers have analyzed and tested several human behaviors for smartphone user authentication. For example, the way a user walks [11], the way a user types (touch-stroke) her secret [2], and how she interacts with the touchscreen [7, 12, 20].

This paper presents a bimodal smartphone user authentication scheme based on two human behaviors - the way a user taps a combination of 10-digits *"text-independent"* number, not necessarily secret (replicating dialing behavior) and the phone's micro-movements during the dialing process (see Figure 1). We chose 10-digits length because we empirically found this length more accurate and resilient compared to 4, 6, 8, 12, and 14 digits. The system profiles the users based on (i) the differences in the tap-timings, and (ii) the phone micro-movements during the entire process of dialing, to authenticate the users. DialerAuth does not require users to remember anything, and hence users can tap any combination of 10-digit number. An attacker has to successfully mimic the two invisible and inherently secure human behaviors, i.e., the invisible timings and the phone micro-movements of the legitimate user, to access the phone.

We registered users' generated phone micro-movements using accelerometer and gyroscope sensors, commonly available on most of the commercially available smartphones. DialerAuth starts these sensors when the user taps the first digit and stops when the user taps the last. We evaluated DialerAuth on our collected dataset of 93 (97, in total) qualified users, by applying anomaly detection (one-class classification) approach. Results show that MLP verifier outperformed other chosen verifiers and performed well in all the postures, i.e., *sitting, standing, walking and walking downstairs*.

The main contributions of this work are:

- The proposal of a bimodal smartphone user authentication scheme - DialerAuth, which authenticates the user based on the differences in the tap-timings while she taps/enters 10-digit *"text-independent"* number, and the phones micro-movements while doing so.
- Prototype implementation of the proposed solution on a smartphone.
- Collection and sharing of collected data set of 93 qualified participants, for further research.
- The security evaluation of DialerAuth to assess its robustness against the most common (random and mimic) attacks.
- The usability evaluation of DialerAuth to assess how users reacted to the solution and estimation of user acceptance.

Paper Organization. The rest of the paper is organized as follows: Section 2 surveys the related work which motivates why there is a need for an improvement. Section 3 explains the threat model and the main technologies used by our solution. Section 4, reports the detailed methodology, i.e., protocols for data collection, feature extraction and selection, analysis, etc., of our conducted experiments. Section 5, explains the proof-of-concept application that we developed based on our findings. Sections 6 and 7 presents the results of security and usability evaluation of dialerauth. Finally, Section 8 concludes the paper with the summary of our findings and future work.

2 RELATED WORK

Behavioral biometrics have increasingly been used for transparent and frictionless user authentication schemes on smartphones. Research on user authentication has been reshaped due to the inclusion of sufficiently sensitive mobile sensors, i.e., accelerometer, gyroscope, touchscreen, etc. As a result, multiple schemes leveraging these sensors have been introduced. Among all, the two behavioral biometric schemes, i.e., the way a user walks (gait) [11, 23], and the way the user interacts with the touchscreen [6, 7, 22, 32]) have dominated the research in the smartphone user authentication domain.

Most of the available smartphones nowadays are equipped with inertial sensors, e.g., accelerometer, gyroscope, etc., which can be used to fingerprint the device movement. This unique movement has explicitly been exploited for smartphone user authentication in recent studies [4]. The proposed solutions are either unimodal [8], bi-modal [6, 9, 14, 16], one-shot [6, 7] and continuous [13, 26].

Since, dialerauth is a bimodal one-shot (i.e., at login stage) authentication scheme which leverages *"text-independent"* tapping behavior and hand micro-movements, we consider sensor-assisted one-shot schemes [6, 7, 14, 16] closer to our work and we compare our work with them.

Ho [16] besides collecting the touch-based data, i.e., hold and dwell time, key-tap size, etc., also investigated the effectiveness of accelerometer readings generated during the process of entering the pre-defined 4-digit PIN. The study involving 55 users showed that motion-based features performed the best as compared to other feature types, while with the fusion of all the feature types, the accuracy was drastically improved (FAR =4.4% and FRR = 5.3%) using SVM as the classifier.

UNAGI, introduced by Giuffrida et al., [14], presents a sensor-enhanced fixed-text scheme for user authentication on Android smartphones. They reported an EER of 4.97% on fixed-text passwords and 0.08% on sensory data. Later, Buriro et al., [6] modified this scheme by introducing the concept of a 4-digit *"text-independent"* secret in addition to the sensory readings. They achieved 1% EER on a dataset of 12 users on fused bimodal data. Similarly, the scheme introduced in [7] leverages the sensory readings while a user writes her name on the smartphone touchscreen. This bimodal system achieved ≈ 95% TAR at 3.1% FAR.

Our proposed scheme dialerauth is different from the existing state of the art in several ways. First, in terms of the input method, all the schemes [6, 14, 16] require text entry using the soft-keyboard, whereas dialerauth uses the dialer for text entry, which is quite important in terms of user experience. Second, our experiments are based on a comparatively large-scale dataset involving 97 users (93 qualified in *sitting*, 90 qualified in *standing* and *walking*, and 72 in *walking downstairs*) activity, while [14], [6], [7], and [16] involved only 20, 12, 30, and 55 users, respectively and testing only partially

the different postures. As the users in these studies were less, it is difficult to assess how their achieved error-rates would have varied, given that the number of users is a very important factor in assessing the confidence on the analysis. Third, all the schemes in [6, 7, 14, 16] performed experimentation in a controlled way, i.e., in a lab and in one session. Whereas, DIALERAUTH leverages the collection of data in multiple sessions and data was collected in totally un-controlled and un-supervised manner.

3 PROPOSED METHOD

3.1 Threat Model

We consider the situation where an attacker is already in possession of the smartphone. This scenario is common because the user might forget her smartphone somewhere, i.e., in her office, canteen, etc., or an attacker manages to steal the smartphone (e.g., through pickpocketing, etc). More specifically, we target three scenarios: (i) an attacker accidentally finds the smartphone, (ii) the attacker is victim's friend or colleague (who knows about the implemented mechanism), and (iii) an attacker who tries to mimic the user behaviors (e.g., using recorded video, etc) to unlock the victim's smartphone.

3.2 Our Approach

DialerAuth requires its users to enter or tap any combination of 10-digits long number to replicate their dialing process and the way the user holds the device in her hand while tapping. We expect that most of the users are familiar with the dialing process. The first authentication factor in DIALERAUTH is how the user enters her *"text-independent"* number. The second authenticating factor is the hold behavior (profiled using the extracted features from the accelerometer and gyroscope sensors). Hence, imitating the two inherently invisible behavioral characteristics, is quite difficult. DIALERAUTH exploits hand micro-movements and the classical keystroke features, i.e., hold and dwell time to authenticate the user. DIALERAUTH captures the key-hold and inter-stroke timings from the 10-digit long number. Similarly it extracts statistical features from the phone's micro-movements to profile the genuine user (as shown in Figure 2). The extracted feature vectors are concatenated together to form a final fused vector to model a bimodal system. Since, the target device is the smartphone, we apply feature selection scheme to (i) improve the system's accuracy, and to (ii) improve the performance (as processing of smaller feature vector may incur less computational cost and processing, making decision time comparatively shorter).

DIALERAUTH also includes the parameter optimization block in which we choose the best parameter, i.e., *number of hidden layers* in MLP. Since, we need to implement the proof-of-concept app, we need to finalize the best features and best parameters. DIALERAUTH creates user templates on the selected features set and trains the verifiers on those templates and saves those templates in the database. Later, the testing template is formed similarly to the training templates and matched with the pre-enrolled templates to authenticate the user.

4 METHODOLOGY

In this section, we discuss the steps taken to design DIALERAUTH.

4.1 Data Collection

We developed a customized Android application, namely *StrokeCollector*, to collect stroke timings and the sensory readings while doing so. Our application can be installed on any Android smartphone having Android 4.4 or higher version. Our app collects sensory readings at 50Hz because this rate was empirically found suitable, for user behavior profiling, in recent studies [6, 7].

We involved "UBERTESTERS[1]" - a crowd-sourcing platform to collect unsupervised field data. "UBERTESTERS" recruited 97 testers and made sure that each tester uses one and always the same smartphone for the whole experiment.

StrokeCollector collected users' interactions in 3 sessions in 3 consecutive days. It required an interaction of 30 minutes on the first day and 15 minutes on the two subsequent days. In this way, each participant had to test the app for 1 hour but including also cool down periods.

4.1.1 Participants. All the recruited testers were application testers and had IT background but not security experts. We set up a webpage, explaining the purpose, the methodology of the experiment, the user consent, and the procedure to install/uninstall the application. Out of 97 recruited users by "UBERTESTERS", some of them were disqualified because of (i) the non-availability of gyroscope sensors, (ii) their samples have Not A Number (NaNs) values, and (iii) the users had less than 30 observations in an activity. Most of the users (72), finished the whole experiment in $40 - 45$ minutes and some of them (25, users) could not finish in their allocated time of 1 hour. To test the impact of different postures in the chosen behavioral traits, the data were collected in 4 different postures (*sitting*, *standing*, *walking*, and *walking downstairs*). The posture was explicitly indicated by the tester during the experiment.

We obtained the usable data of 93 participants in *sitting*, 90 in *standing* and *walking* and 72 users in *walking downstairs* activities. The volunteers were required to install the application, answer to the demographic questions, and perform dialing actions 30 times in four activities. In total, we collected 120 samples from each user in all four activities in first two days. On the third day, the previously collected samples were used to train the dummy classifier and users had to test the app by 30 testing samples in any activity as per their preference. After each testing attempt, the result of the dummy classifier (either accept or reject) was shown as the toast message to give the tester an idea of the scheme. Since, we are not sure about the activity of testing, we do not use this data, in this paper.

Our demographics questionnaire included 4 questions. Volunteers were free to answer those or choose if they did not want to disclose. The demographics questions, and options to answer these questions, are listed in the appendix A and the summary of collected dataset and corresponding demographics is tabulated in Table 1.

Lastly, the users' had to record their experience with our scheme using System Usability Scale (SUS)[2] questionnaire. All the users who completed the whole experiment filled in the SUS questionnaire embedded in our app. Additionally, we sent a link to all the remaining participants to know their opinion about our scheme.

[1] https://ubertesters.com/
[2] https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

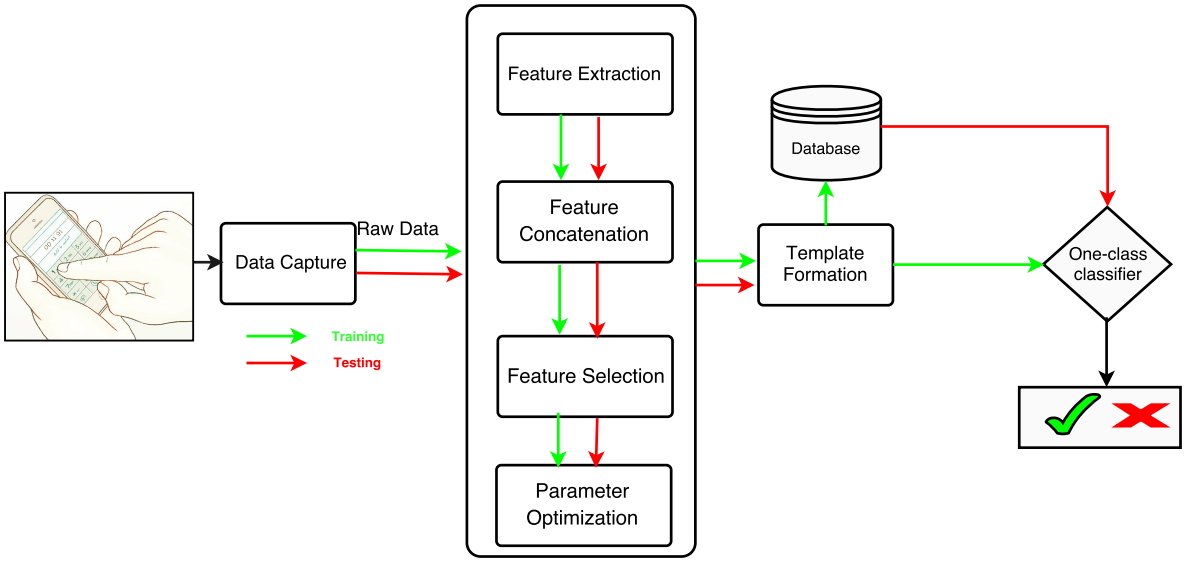


Figure 2: Model diagram of DIALERAUTH.

All the participants provided their feedback and so our usability analysis is based on the feedback of all 97 users.

Table 1: User demographics (M = Male, F = Female, U = Undisclosed, R = Right, L = Left, B = Both)

Information	Description
No. of Users	93 qualified (100, in total)
Sample Size	**Sitting** 2790 (93 **X** 30) **Standing** 2700 (90 **X** 30) **Walking** 2700 (90 **X** 30) **Downstairs** 2, 160 (72 **X** 30)
Devices	Android Smartphones with 4.4.x version
No. of Sessions	3
Gender	77(M), 20(F), 0 (U)
Handedness	91(R), 6(L) 0 (U)
Age Groups	92 (20 – 40), 5 (41 – 60)

4.2 Feature Extraction

Accelerometer and gyroscope are both 3-dimensional sensors commonly available on all the smartphones available today. We also calculated the magnitude (like the previous studies [6, 7]) using the following equation:

$$S_M = \sqrt{(a_x^2 + a_y^2 + a_z^2)} \quad (1)$$

where S_M is the magnitude and a_x, a_y and a_z are the accelerations along the X, Y and Z directions.

All the four dimensions are processed to extract the following statistical features: Mean, Standard Deviation (STD), Skewness, Kurtosis, from each of the dimension. In this way, 16 statistical features were extracted from each sensor (see Table 2). The final feature vector of hold behavior is 32 features long, formed by concatenating the features from two chosen sensors: accelerometer and gyroscope. Similarly, dialing behavior is profiled using the 38 timing-based features (dwell and flight time), as shown in Figure 3.

Table 2: List of selected features from sensory readings.

No.	Lift Features			
1-4	MeanX	MeanY	MeanZ	MeanM
5-8	STDX	STDY	STDZ	STDM
9-12	SkewX	SkewY	SkewZ	SkewM
13-16	KurtX	KurtY	KurtZ	KurtM

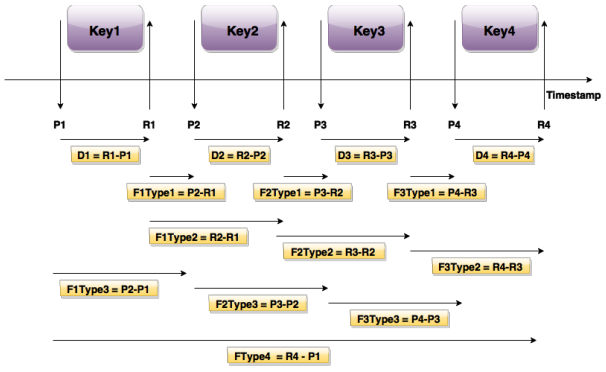


Figure 3: Features of dialing adapted from [6](for 4 digits).

4.3 Feature Fusion

The acquired biometric-data can be fused at 5 different stages, namely, sensor, feature, score, rank, and decision. The literature recommends the fusion of data as early as possible to obtain higher accuracy [17]. We preferred feature fusion over sensor level because at sensor level presence of noise could have affected, significantly, the recognition accuracy. Since, both the modalities are independent of each other, we concatenated the features of hold and dialing behavior, as suggested in [25], to form a final feature vector of 70 features to profile the user.

4.4 Feature Subset Selection

Feature subset selection is the process of selecting the most productive feature subset (based on higher accuracy) out of the full feature set. We considered feature selection because processing smaller feature vector incurs less computational time and cost. Additionally, smaller vector makes easier for the verifier to learn quickly as compared to learning on the redundant and irrelevant features.

We evaluated our combined feature set (70 features long) with a Weka-based feature selection scheme - the Information Gain Attribute Evaluator (IGAE) - a mutual-information based feature selection scheme, which ranks the features based on their contained information gain with respect to the class. The outcome of this scheme provides the feature ranking based on their feature weights (higher the better). We computed the threshold by taking the ratio of the number of features (i.e., 70) to the number of users per activity. For example, for sitting activity, the computed threshold was $70/93 = 0.75$. The features above this threshold are taken for further analysis. The list of selected features for all the activities is shown in Table 3. Motion-based features have dominated the touch-based features (see Table 3) because the touch-based features are computed on typing of arbitrary keys (independent of the content of the key) and this makes them a bit weak.

Table 3: List of selected features from fused (bi-modal) data.

Sitting	*Standing*	*Walking*	*Downstairs*
Acc_MeanX	Acc_MeanX	Acc_MeanX	Acc_MeanMag
Acc_MeanY	Acc_MeanY	Acc_MeanY	Acc_MeanX
Acc_MeanZ	Acc_MeanZ	Acc_MeanZ	Acc_MeanY
Acc_MeanMag	Acc_MeanMag	Acc_MeanMag	Acc_MeanZ
Acc_SkewX	Acc_SkewX	Acc_SkewX	Acc_STDMag
Acc_SkewZ	Acc_SkewZ	Acc_SkewZ	Acc_STDY
Acc_SkewMag	Acc_SkewMag	Acc_SkewMag	Acc_STDZ
Gyro_MeanX	Gyro_MeanX	Gyro_MeanX	Gyro_MeanMag
Gyro_MeanMag	Gyro_MeanMag	Gyro_MeanMag	Gyro_STDZ
Gyro_SkewX	Gyro_SkewX	Gyro_SkewX	FType4
Gyro_SkewY	Gyro_SkewY	Gyro_SkewY	-
Gyro_MeanZ	Gyro_MeanZ	Gyro_MeanZ	-
Gyro_SkewMag	Gyro_SkewMag	Gyro_SkewMag	-
-	-	F4Type2	-
-	-	F4Type3	-

4.5 Verifier Selection

We consider user authentication as one-class classification (anomaly detection) problem [7, 26], where the data from the "+ve" class (owner), is used to train the system and is tested against the data from all other "-ve" (non-owners) classes. Trained verifier checks for the deviation between the testing sample to the training samples and decides, accordingly. Less different samples (with smaller deviation) are accepted as the patterns belonging to the "owner" and vice versa.

We used Weka - a gui-based workbench to perform one-class classification. We used multiple verifiers, namely, Bayesian Networks, K-Nearest Neighbors, Multilayer Perceptron (MLP) and Random Forest (RF), wrapped into the Weka meta-classifier -the OneClassClassifier[3].

Bayesian classifiers such as Belief Networks (BN) use probabilistic techniques for classification. These classifiers/verifiers are widely used because of their super simplicity and faster learning capability. K-Nearest Neighbor (KNN) classifier is a density based simple and fast learner classifier [28] which can perform pretty well on any kind of a dataset and thus was considered among the top 10 classification algorithms [30]. Random Forest (RF) classifier grows multiple classification trees and classifies by providing the input vectors to each of the tree and infers the label of the class based on the majority voting. RF classifier learns quickly and never overfits [3], however it requires more memory compared to other classifiers. Multilayer Perceptron (MLP) classifier, belongs to the neural network family, and is composed of different layers, i.e., input layer, hidden layer, and output layer. It trains a non-linear approximator by learning on the feature vectors and their corresponding labels. It is found extremely useful classifier in recent studies [7, 8].

4.6 Parameter Optimization

For designing an MLP neural network, the question *"how many hidden layers?"* is very important and the solution depends mainly on the characteristics of the dataset, e.g., is the data linearly separable, etc. For linearly separable dataset, the default settings, i.e., one hidden layer, could provide the optimum results. However, because of the non-linearity of the data, we tried a number of hidden layers between $1 to 10$ to find out the best parameter. We achieved highest TAR with 9 hidden layers. It should be noted that we performed this evaluation on selected IGAE features and used the same protocol (i.e., owner vs all would-be attackers) for evaluating different parameters for the MLP verifier.

4.7 Analysis

Our analysis starts by profiling one of the testers as the "owner" and the remaining users as "non-owners". We applied 5-fold cross-validation on the data from "owner" class to obtain the True Acceptance Rate (TAR) and False Rejection Rate (FRR). TAR shows the rate of correctly accepted patterns, and FRR shows the rate of incorrectly rejected patterns, of the genuine users, i.e., "owner". Then, we train the verifiers on the data of the genuine user, i.e., "owner" and test it against the data of all the remaining "non-owner" classes. The result of this setting also provides the binary output, i.e., False Accept Rate (FAR) and True Reject Rate (TRR). FAR shows the rate of incorrectly accepted impostor patterns as the genuine

[3] http://weka.sourceforge.net/doc.packages/oneClassClassifier/weka/classifiers/meta/OneClassClassifier.html

users patterns and TRR shows the correctly rejected impostors patterns. We repeated the testing for all the users and report the average results.

4.8 Results

We show the results of our chosen verifiers in terms of $TAR = 1 - FRR$, and $FAR = 1 - TRR$. We do not report FRR and TRR to avoid the redundancy. Table 4 summarizes the obtained results (with the default settings of the verifiers) with full feature sets prior to the feature selection. Since MLP performed well yielding comparatively better results (and defeating KNN in 3 activities), we take this verifier for further analysis.

Table 4: : Results of different classifiers on full (70) and features (averaged over 93 (*sitting*), 90 (*standing & walking*) and 72 (*downstairs*) participants) in different activities.

	Sitting		*Standing*		*Walking*		*Downstairs*	
Classifiers	TAR	FAR	TAR	FAR	TAR	FAR	TAR	FAR
BN	69.37	12.63	67.83	11.69	67.21	8.76	67.96	13.97
RF	70.05	12.22	68.79	11.34	68.21	8.48	68.40	13.58
MLP	**71.84**	12.63	**71.05**	11.66	**69.87**	8.71	**70.59**	13.89
KNN	**72.11**	12.61	**70.38**	11.76	**68.45**	8.81	**69.74**	14.23

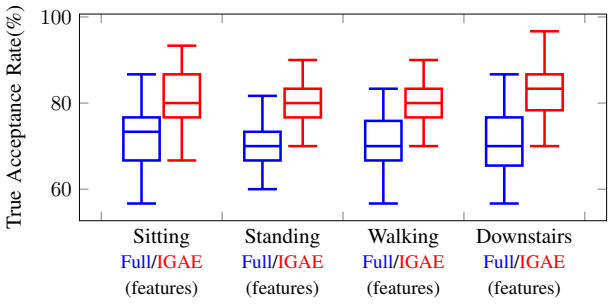


Figure 4: Comparison of TAR for full and extracted features.

With the most accurate verifier, i.e., MLP with full features, we obtained 69.87% to 71.84% TAR (see Table 4) in 4 different activities. The performance of the MLP verifier was increased on the selected IGAE features (see Figure 4). It is evident from the figure that the patterns of each of the participant are discriminated with higher accuracy. With the default parameters of MLP, i.e., with just one hidden layer, and using IGAE selected features, the achieved TAR is 81.63% compared to 71.84% (sitting), 81.13% compared to 71.05% (standing), 79.34% compared to 69.87% (walking), and 81.63% compared to 70.59% (walking downstairs). Furthermore, by applying the parameter optimization and using IGAE selected features, the TAR was improved to a significant level, i.e., from 71.84% to 85.77% in *sitting*, from 71.05% to 85.02% in *standing*, from 69.87% to 84.69% in *walking*, and from 70.59% to 84.36% in *walking downstairs*, activity.

Since we achieved the highest TAR for three activities with 9 hidden layers (see Figure 5) and lowest FAR with 8 hidden layers (see Figure 6), however, due to very less difference between the observed FAR with 8 and 9 layers, we prototype final implementation with 9 hidden layers.

We accept that the overall performance of the system is less at this stage, compared to the sensor-enhanced fixed-text keystroke dynamics [14] however, it can be improved with the passage of time. This feature of dialing any arbitrary keys makes DIALERAUTH more usable because a user does not need to memorize the code.

5 PROOF-OF-THE-CONCEPT IMPLEMENTATION

Evaluating the accuracy of DIALERAUTH for intra-activity (training and testing in the same activity) is not enough. In real world situations, the user uses the smartphone in multiple situations, i.e., walking, biking, cycling, etc. So if the DIALERAUTH is trained in one activity, it will be useful only in that particular activity [7]. There are two solutions to this problem: firstly, the training needs to be done in multiple possible activities and their corresponding patterns needs to be saved separately in the database. During testing first the activity needs to be estimated using activity estimator (i.e., JigSaw [19]) and after determining the activity, the patterns needs to be compared with the pre-enrolled patterns of the corresponding activity. In this way, users have to train the system very well and it requires more training and end up in asking more training time and effort from the user. Additionally, this approach delays the decision time when the user logs in. Alternatively, as proposed in [7], multiple activities can be fused as well. For example, training the system with just 50 patterns in all possible activities could lead both to the user acceptance and the performance improvement. Similarly, the activities can be fused by creating a common feature vector based on the selected features, i.e., most of the features are common across activities (see Table 3).

The final proof-of-concept implementation of DIALERAUTH is based upon our findings discussed in the previous sections. It is worth repeating that MLP verifier performed well both on the full features, and selected features and provided highest TAR with 9 hidden layers. Thus, DIALERAUTH uses MLP with 9 hidden layers and the selected features (listed in Table 3) to authenticate the user. DIALERAUTH can be installed on any smartphone having in-built accelerometer and gyroscope sensors and running any Android version 4.4 or higher. DIALERAUTH requires minimal configuration at the time of installation. DIALERAUTH helps its user by displaying the suggested recommended value. At the end, the user is required to either only tap, or move (in the case of training over one modality) or do both to get authenticated.

6 SECURITY ANALYSIS

6.1 Considerations

DIALERAUTH cannot be easily attacked by a random attacker. The attacker can mount a side-channel attack using motion sensors [21], i.e., the attack can recover the entered text but not the stroke-timings. Therefore, despite having the knowledge of the entered text, the attacker would not be able to mimic the manner in which the text was entered.

As other keystroke-based schemes, DIALERAUTH too relies upon hidden features (i.e., timing-differences), it becomes than extremely hard to mount shoulder surfing, mimic attacks (through recorded

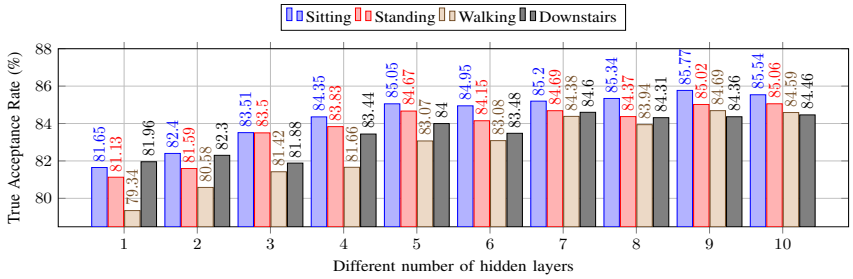


Figure 5: Parameter Optimization: highest TAR achieved is 85.77% **with 9 hidden layers in sitting,** 85.02% **in standing,** 84.69% **in walking , and with 7 hidden layers** 84.6% **in walking downstairs. Obtained results are averaged over** 93 **(*sitting*),** 90 **(*standing*),** 90 **(*walking*), and** 72 **(*downstairs*) users.**

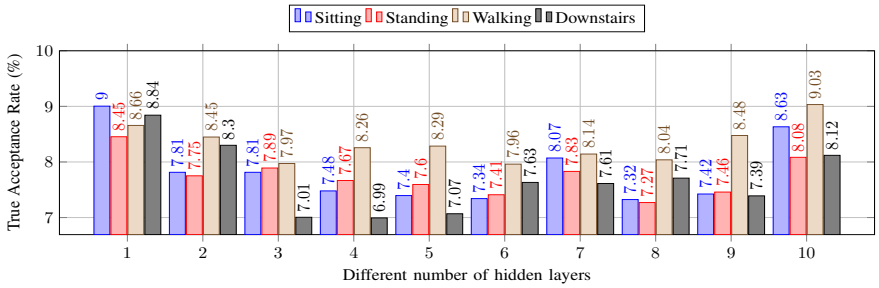
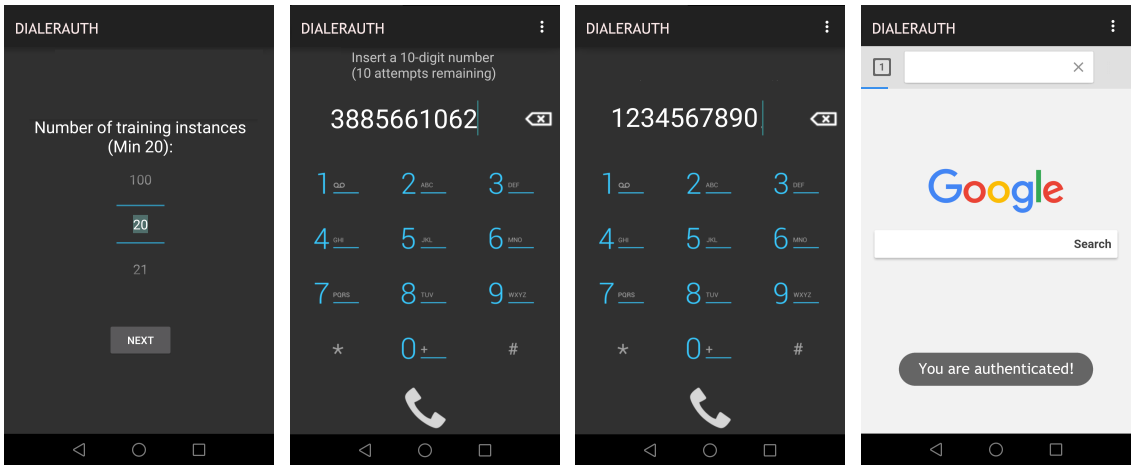


Figure 6: Parameter Optimization: Lowest FAR achieved is 7.32% **with 8 hidden layers in sitting,** 7.27% **in standing,** 8.04% **in walking , and with 4 hidden layers** 6.99% **in walking downstairs. Obtained results are averaged over** 93 **(*sitting*),** 90 **(*standing*),** 90 **(*walking*), and** 72 **(*downstairs*) users.**



(a) (b) (c) (d)

Figure 7: (a) & (b) Training the system, (c) & (d) testing the system.

videos, etc.), and other synthetic approaches [27]. Further DIALERAUTH also depend upon the invisible phone's micro-movements the user makes while dialing. These movements are difficult to capture (at least using just naked eyes) and depend on various person-specific attributes such as height, weight, and length of the arm. Hence mimicking these micro-movements could require to mount a quite expensive attack (i.e., using computer vision techniques). If that is considered a plausible attack, using a secret number rather than any number can be a way to mitigate such attack.

Table 5: Results of SUS-based usability study.

SUS Questions	*Responses*				
	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
I think that I would like to use DIALERAUTH frequently	3	7	39	30	19
I found DIALERAUTH unnecessarily complex	34	21	28	10	4
I thought DIALERAUTH was easy to use	1	4	14	34	44
I think that I would need the support of a technical person to be able to use DIALERAUTH	59	18	11	7	2
I found the various functions in DIALERAUTH were well integrated	2	5	24	44	22
I thought there was too much inconsistency in DIALERAUTH	28	24	27	15	3
I would imagine that most people would learn to use DIALERAUTH very quickly	2	5	11	39	40
I found DIALERAUTH very cumbersome to use	28	21	31	12	5
I felt very confident using DIALERAUTH	1	5	14	36	41
I needed to learn a lot of things before I could get going with DIALERAUTH	56	21	11	8	1

6.2 Evaluation

In order to empirically determine the robustness of DIALERAUTH against possible attacks, we recruited 8 more testers (would-be adversaries) and performed the additional experiments. We installed the prototype application on three smartphones, i.e., HUAWEI GRA-L09, Samsung OnePlus One, and LG-Flex 2, trained their classifiers with the data of 3 randomly chosen participants of the main experiment. After the random attacks, the classifiers of these devices were re-trained in front of the testers with the intention that the testers would learn well the legit user's dialing behavior and attempt mimicking the trained behavior, effectively. We explained the complete experiment to the participants and requested them to dial any 10-digit number to impersonate the legit user. For each adversarial attempt, the trained classifier checks for the similarity between the testing and training samples and shows the decision as a Toast message on the screen. The outcome of the classifier was binary: authenticated or rejected and we save these outcomes as FAR and TRR.

6.2.1 Random Attacks. In this scenario, we requested the test-adversaries to dial any combination of 10-digit number and press enter to break the login process on the pre-trained classifier (over 30-dialing patterns of the legitimate user). The process was repeated for 3 legit users. Each would-be adversary tried 5 times on each smartphone (15 on all 3 smartphones and 120, in total, from all 8 testers). We tested with 5 adversarial attempts because Android devices after 5 unsuccessful attempts lock themselves for some time. All of the 120 adversarial attempts went unsuccessful, i.e., none of the attempt could break the login process.

6.2.2 Mimic Attacks. In this scenario, a valid user was asked to mimic the behaviors of the 3 legitimate users. The legit users trained their smartphones by dialing 10 times in three activities, i.e., *sitting*, *standing*, and *walking*, respectively, in front of the would-be attackers and later they were required to mimic the behaviors of the valid user. Some of the would-be adversaries also needed more demonstrations and practice and were allocated their required time slot. All 8 test adversaries were asked to mimic real user's behaviors in 5 attempts (15 total, on 3 smartphones) in any of their preferred activity. 7 out of the 120 adversarial attempts went successful. It is worth-mentioning that 6 out of 7 successful attempts succeeded to spoof one legit user compared to 1 and 0 for other two legitimate users. The victim legitimate user might belonged to the "lambs[4]" family [31], however, majority of the users tend to be "sheeps[5]".

7 USABILITY ANALYSIS

We admit that the use of dialing process - user's action of entering telephone numbers, might be difficult for the enrollment to some, however, because of the (i) users' familiarity with the dialing process, and (ii) the flexibility to dial 10-digit *"text-independent"*, DIALERAUTH can get wide user acceptance. To assess our intuition related to usability, we relied on SUS to gather the testers reviews about DIALERAUTH. SUS is a 10-questions based questionnaire and considered as an standard tool to evaluate the usability of any system.

7.1 Methodology

The SUS[6] is a standard tool widely used to record user impressions about the usability of a system and has been also used in the context of smartphone user authentication [7, 22, 29]. The users' response to each question is recorded on a given 5-point scale ranging from "Strongly Disagree" to "Strongly Agree" as shown in Table 5. The outcome is computed as a score between 0 and 100 and termed as SUS score. The obtained score, x, is categorized as follows: (i) Best Imaginable ($x \geq 92$), (ii) Excellent ($92 < x \geq 85$), (iii) Good ($85 < x \geq 72$), (iv) OK ($72 < x \geq 52$), (v) Poor ($52 < x \geq 38$), and (vi) Worst Imaginable ($38 < x \geq 25$). We replaced the word "System" with 'DIALERAUTH'. In addition to the standard SUS questions, we also added some questions to know about testers' background and their feedback (see Appendix B).

[4] "Lambs are vulnerable to impersonation. When being matched against, they result in relatively high match scores, leading to potential false accepts."
[5] "Sheep make up the majority of the population of a biometric system. On average, they tend to match well against themselves and poorly against others".
[6] https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

7.2 Responses

We report the usability analysis of DIALERAUTH on the basis of the feedback we received from all 97 study participants. The summary of the obtained responses is presented in Table 5. It is evident from the table (Question 1) that 49 users (≈ 51%) were agreed or strongly agreed to use DIALERAUTH compared to just 10 (10%) who disagreed or strongly disagreed. On the simplicity and user-friendly point (Question 3), 78 (80%) of the users considered DIALERAUTH easy to use. Figure 8 illustrates the break up of obtained SUS score. Significant fraction of users (19.58%) rated DIALERAUTH as the best imaginable authentication solution, and 10.30% and 28.86% of users considered it as Excellent and Good, respectively. Overall, DIALERAUTH was well accepted by the testers and it obtained a mean score of 73.29 (mean SUS score of 72 or higher is considered better with the higher probability of wide user acceptance).

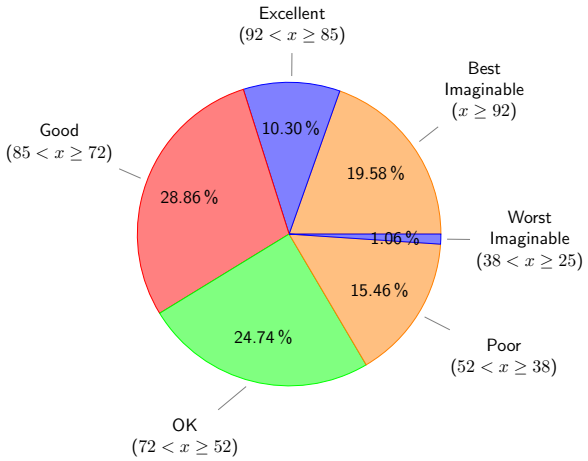


Figure 8: Break-up of the obtained SUS score.

In response to the question: "*How long have you been using the smartphone?*", 95 of 97 (≈ 98%) users said they have been using it for more than 6 months. We assumed that most of the smartphone users could become the expert users after 6 months of regular usage.

Further, in response to the question "*Would you consider to use DIALERAUTH to replace the authentication mechanism you are using?*", ≈ 48% users replied with "Yes". More specifically, 75% of the users who did not have any authentication mechanism enabled, 38.46% slide-to-unlock users, 54.54% PIN/password users, 42.85% biometrics users, and 80% of the smartlock users, answered with "YES". This is positive indication of the user acceptance of DIALERAUTH.

As per the recorded feedback, most of the users looked satisfied mentioning DIALERAUTH as a simple, extremely convenient, user-friendly and intuitive authentication mechanism. The flexibility of tapping/dialing any number made users more comfortable.

We also got some negative responses related to the length of the digits and training the scheme. The scheme requires a fair amount of training patterns, i.e., 30 (as the users have to dial or tap multiple times for training compare to setting up a PIN or registering the face) and hence becomes unacceptable to some.

We will incorporate these suggestions into future version of DIALERAUTH before its deployment to the real world.

8 CONCLUSION AND FUTURE WORK

We have proposed a simple, secure and user-friendly, smartphone user authentication scheme leveraging the user's familiarity with the dialing process. The proposed scheme authenticates the users based on the timing differences in successive strokes while user taps 10-digit *"text-independent"* number. DIALERAUTH further adds an invisible and unobservable layer, based on the hand's micro-movement, to strengthen the security of the scheme. Hence, an attacker has to successfully mimic the two invisible and inherently secure human behaviors, i.e., the invisible timings and the phone micro-movements of the legitimate user, to access the phone.

DIALERAUTH counters the most common problems associated with different authentication schemes, e.g., user does not need to remember any secret (thus eliminating the problem of sharing and stolen passwords), it does not require any additional hardware (as the biometric technology, e.g., iris, fingerprint requires expensive dedicated hardware) and makes spoofing very difficult (as mimicking two invisible behaviors requires a lot of effort and time).

As a future work, we plan to check the effectiveness of DIALERAUTH towards continuous authentication. We are also going to address the problem of seamless and fast detection of a user's current activity [5, 7], i.e., using Jigsaw [19], since this would allow authenticating users based on the best feature subset selected from that particular activity. We are also going to solve the DIALERAUTH training problem (as its training in one go could be extremely cumbersome for some users) by collecting the dialing behavioral patterns in daily usage and after reaching a best number (e.g., 30), DIALERAUTH would notify the user about the availability of this modality for authentication.

9 ACKNOWLEDGEMENT

The authors would like to thank all the participants of the experiment for their time and effort, colleagues for valuable and insightful input and anonymous reviewers for their reviews and comments.

The work was partially supported by the European Training Network for CyberSecurity (NeCS) grant number 675320.

REFERENCES

[1] Zahid Akhtar. 2012. *Security of multimodal biometric systems against spoof attacks.* Ph.D. Dissertation. Department of Electrical and Electronic Engineering, University of Cagliari, Cagliari, Italy. Advisor(s) Fabio Roli.

[2] Francesco Bergadano, Daniele Gunetti, and Claudia Picardi. 2002. User authentication through keystroke dynamics. *ACM Transactions on Information and System Security (TISSEC)* 5, 4 (2002), 367–397.

[3] Leo Breiman and Adele Cutler. 2018. Random Forests. (2018). Retrieved January 8, 2018 from https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#features/

[4] Attaullah Buriro. 2017. *Behavioral Biometrics for Smartphone User Authentication.* Ph.D. Dissertation. University of Trento, Italy. Advisor(s) Bruno Crispo.

[5] Attaullah Buriro, Bruno Crispo, Filippo Del Frari, Jeffrey Klardie, and Konrad Wrona. 2015. Itsme: Multi-modal and unobtrusive behavioural user authentication for smartphones. In *International Conference on Passwords.* Springer, 45–61.

[6] Attaullah Buriro, Bruno Crispo, Filippo Del Frari, and Konrad Wrona. 2015. Touchstroke: smartphone user authentication based on touch-typing biometrics. In *International Conference on Image Analysis and Processing.* Springer, 27–34.

[7] Attaullah Buriro, Bruno Crispo, Filippo Delfrari, and Konrad Wrona. 2016. Hold and Sign: A Novel Behavioral Biometrics for Smartphone User Authentication. In *IEEE Security and Privacy Workshops (SPW).* 276–285.

[8] Attaullah Buriro, Bruno Crispo, and Yury Zhauniarovich. 2017. Please Hold On: Unobtrusive User Authentication using Smartphone's built-in Sensors. In *IEEE International Conference on Identity, Security and Behavior Analysis (ISBA-2017)*.
[9] Attaullah Buriro, Sandeep Gupta, and Bruno Crispo. 2017. Evaluation of Motion-based Touch-typing Biometrics for online Banking. (2017).
[10] Alexander De Luca, Alina Hang, Emanuel Von Zezschwitz, and Heinrich Hussmann. 2015. I Feel Like I'm Taking Selfies All Day!: Towards Understanding Biometric Authentication on Smartphones. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 1411–1414.
[11] Mohammad Omar Derawi, Claudia Nickel, Patrick Bours, and Christoph Busch. 2010. Unobtrusive user-authentication on mobile phones using biometric gait recognition. In *Proceedings of the 6th IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*. 306–311.
[12] Mario Frank, Ralf Biedert, Eugene Ma, Ivan Martinovic, and Dawn Song. 2013. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *IEEE transactions on information forensics and security* 8, 1 (2013), 136–148.
[13] Hugo Gascon, Sebastian Uellenbeck, Christopher Wolf, and Konrad Rieck. 2014. Continuous Authentication on Mobile Devices by Analysis of Typing Motion Behavior.. In *Sicherheit*. 1–12.
[14] Cristiano Giuffrida, Kamil Majdanik, Mauro Conti, and Herbert Bos. 2014. I sensed it was you: authenticating mobile users with sensor-enhanced keystroke dynamics. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 92–111.
[15] Marian Harbach, Emanuel Von Zezschwitz, Andreas Fichtner, Alexander De Luca, and Matthew Smith. 2014. It's a hard lock life: A field study of smartphone (un) locking behavior and risk perception. In *Symposium on usable privacy and security (SOUPS)*. 213–230.
[16] Grant Ho. 2014. *Tapdynamics: strengthening user authentication on mobile phones with keystroke dynamics*. Technical Report. Stanford University.
[17] Anil Jain, Arun A Ross, and Karthik Nandakumar. 2011. *Introduction to biometrics*. Springer Science & Business Media.
[18] Markus Jakobsson, Elaine Shi, Philippe Golle, and Richard Chow. 2009. Implicit authentication for mobile devices. In *Proceedings of the 4th USENIX conference on Hot topics in security*. 9–9.
[19] Hong Lu, Jun Yang, Zhigang Liu, Nicholas D Lane, Tanzeem Choudhury, and Andrew T Campbell. 2010. The Jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM conference on embedded networked sensor systems*. 71–84.
[20] Yuxin Meng, Duncan S Wong, Roman Schlegel, et al. 2012. Touch gestures based biometric authentication scheme for touchscreen mobile phones. In *International Conference on Information Security and Cryptology*. Springer, 331–350.
[21] Emiliano Miluzzo, Alexander Varshavsky, Suhrid Balakrishnan, and Romit Roy Choudhury. 2012. Tapprints: your finger taps have fingerprints. In *Proceedings of the 10th ACM international conference on Mobile systems, applications, and services*. 323–336.
[22] Toan Van Nguyen, Napa Sae-Bae, and Nasir Memon. 2017. DRAW-A-PIN. *Computers and Security* 66, C (2017), 115–128.
[23] Claudia Nickel, Tobias Wirtl, and Christoph Busch. 2012. Authentication of smartphone users based on the way they walk using k-nn algorithm. In *Proceedings of the 8th IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*. 16–20.
[24] Mudassar Raza, Muhammad Iqbal, Muhammad Sharif, and Waqas Haider. 2012. A survey of password attacks and comparative analysis on methods for secure authentication. *World Applied Sciences Journal* 19, 4 (2012), 439–444.
[25] Arun Ross and Anil Jain. 2003. Information fusion in biometrics. *Pattern recognition letters* 24, 13 (2003), 2115–2125.
[26] Zdenka Sitova, Jaroslav Sedenka, Qing Yang, Ge Peng, Gang Zhou, Paolo Gasti, and Kiran Balagani. 2015. Hmog: A new biometric modality for continuous authentication of smartphone users. *arXiv preprint arXiv* 1501 (2015).
[27] Deian Stefan, Xiaokui Shu, and Danfeng Daphne Yao. 2012. Robustness of keystroke-dynamics based biometrics against synthetic forgeries. *computers & security* 31, 1 (2012), 109–121.
[28] Saravanan Thirumuruganathan. 2010. A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm. (May 2010). Retrieved Nov 20, 2017 from https://goo.gl/qvgpwc
[29] Shari Trewin, Cal Swart, Larry Koved, Jacquelyn Martino, Kapil Singh, and Shay Ben-David. 2012. Biometric authentication on a mobile device: a study of user effort, error and task disruption. In *Proceedings of the 28th ACM Annual Computer Security Applications Conference (ACSAC 2012)*. 159–168.
[30] Xindong Wu, Vipin Kumar, J Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J McLachlan, Angus Ng, Bing Liu, S Yu Philip, et al. 2008. Top 10 algorithms in data mining. *Knowledge and information systems* 14, 1 (2008), 1–37.
[31] Neil Yager and Ted Dunstone. 2010. The biometric menagerie. *IEEE transactions on pattern analysis and machine intelligence* 32, 2 (2010), 220–230.
[32] Heng Zhang, Vishal M Patel, Mohammed Fathy, and Rama Chellappa. 2015. Touch gesture-based active user authentication using dictionaries. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*. 207–214.

Appendices

A DEMOGRAPHIC QUESTIONNAIRE

(1) What is your gender?
- Male
- Female
- I don't want to disclose

(2) How old you are?
- ≤ than 20 years.
- > 20 years and ≤ 40 years.
- > 40 years and ≤ 60 years.
- > than 60 years.
- I don't want to disclose

(3) Tell us about your nationality.
- _____
- I don't want to disclose

(4) Which hand(s) do you use for interacting with your smartphone?
- Right
- Left
- Both
- I don't want to disclose

B ADDED QUESTIONS TO SUS QUESTIONNAIRE

(1) How long have you been using the smartphone?
- ≤ than 6 months.
- > than 6 months.

(2) Which authentication mechanism are you using on your smartphone?
- PIN/Password/Graphical Password
- Android Smartlock
- Biometrics (Face, Fingerprint, IRIS, Voice, etc)
- Slide-to-Unlock
- None

(3) Would you consider to use DialerAuth to replace the authentication mechanism you are using?
- Yes
- No

(4) Do you have any feedback you like to share with us?
- _____