Touchstroke: Smartphone User Authentication Based on Touch-Typing Biometrics

Attaullah Buriro[1(✉)], Bruno Crispo[1,2], Filippo Del Frari[1], and Konrad Wrona[3]

[1] Department of Information Engineering and Computer Science, University of Trento, Trento, Italy
{attaullah.buriro,bruno.crispo,filippo.delFrari}@unitn.it
[2] DistrNet, KU Leuven, Leuven, Belgium
bruno.crispo@cs.kuleuven.be
[3] NATO Communications and Information Agency, The Hague, Netherlands
konrad.wrona@ncia.nato.int

Abstract. Smartphones are becoming pervasive and widely used for a large variety of activities from social networking to online shopping, from message exchanging to mobile gaming, to mention just a few. Many of these activities generate private information or require storing on the phone user credentials and payment details. In spite of being so security and privacy critical, smartphones are still widely protected by traditional authentication mechanisms such as PINs and passwords, whose limitations and drawbacks are well known and documented in the security community. New accurate, user-friendly and effective authentication mechanisms are required. To this end, behavior-based authentication has recently attracted a significant amount of interest in both commercial and academic contexts.

This paper proposes a new bi-modal biometric authentication solution, *Touchstroke*, which makes use of the user's hand movements while holding the device, and the timing of touch-typing(Touch-typing is the act of typing input on the touchscreen of a smartphone.) when the user enters a *text-independent* 4-digit PIN/password. We implemented and tested the new biometrics in real smartphones. Preliminary results are encouraging, showing high accuracy. Thus, our solution is a plausible alternative to traditional authentication mechanisms.

Keywords: Smartphone · Behavioral biometrics · Keystroke · Transparent

1 Introduction

Smartphones and tablets have become essential devices in the lives of many people. A key factor of such success is their ability to offer mobility, computing power, storage capacity and an easy-to-use interface. This combined with the availability of millions of mobile applications explains the huge popularity of such devices.

V. Murino et al. (Eds.): ICIAP 2015 Workshops, LNCS 9281, pp. 27–34, 2015.
DOI: 10.1007/978-3-319-23222-5_4

Access to modern smartphones is still protected by old-fashioned mechanisms such as passwords and PINs. These methods are not only a burden to use, they are also not very secure (susceptible to guessing, shoulder surfing and smudge attacks [1]). Users often leave devices without any protection or choose too-easy-to-guess passwords (e.g. all zeros).

Recently, researchers proposed the use of behavior-based authentication means such as gait, phone movement, touch and keystroke as a replacement for passwords. Behavioral biometrics require minimal interaction during the authentication process, resulting in a significant increase in user acceptability.

This paper presents a new behavior-based authentication scheme called *Touchstroke*, which takes into account two human behaviors: how the phone is held and how a 4-*digit text-independent* PIN/password is entered. Our experiments confirmed that every user has a unique phone movement behavior and a different way of touch-typing a PIN/password on the smartphone. *Touchstroke* computes the phone-holding behavior with 7 built-in smartphone sensors, for: the orientation, the gravity, the magnetometer, the gyroscope and 3 variants of the accelerometer. Sensors are started at the time of the first touch-type and stopped after the fourth and final touch-type. Users are allowed to input any combination of 4-digit numbers and/or alphabets, hence they are expected to be able to use this authentication mechanism quite comfortably.

We extracted 4 statistical features from each data stream from all the physical sensors (a total of 16 from each sensor) and 14 features related with *n-graph*, related with dwell time and flight time (see Figure 1), from each typing pattern. In [2] authors show that these time-based features are the most widely used features in keystroke dynamics. In order to check the usability of our proposed method, we collected 30 observations from 12 users in six significantly different activities. As user authentication is essentially a binary class classification problem, we tested our dataset using two state-of-the art binary classifiers, BayesNET and Random Forest. The reason behind this selection is that they have shorter computation time and resistance against over-fitting.

The remainder of the paper is as follows. Section 2 covers related work. Section 3 explains the sensors and classifiers used. In section 4, we present an initial assessment of our intuition. Section 5 presents the experimental setup, data collection and feature extraction and discusses obtained results. Section 7 presents planned future work and concludes the paper.

2 Related Work

Keystroke-based user authentication is the most evaluated and tested behavioral biometric method for user authentication on PCs and smartphones using hardware and software keyboards. Since we have implemented text-independent touch-typing dynamics using Android software-keyboards, we consider software-keyboard-based work as our related work.

2.1 Software Keyboard-Based User Authentication

Keystroke-based recognition systems use the measurement of user's typing behavior on digital input devices such as smartphones and tablets. A digital signature is prepared on the basis of a user's interactions with these devices. Specifically, a user is asked to provide an alpha-numeric PIN/password to the system for creating a template for training and later for testing. [2,3] suggest that this fingerprinting is fairly unique from person to person thus can be used as a base for user identification.

A study conducted by Huang et al. [4] explored software-based-keyboard user authentication on mobile phones. Users were asked to enter their names and passwords 6 times for training. Based on the keystroke latency and key-hold-time features, the study reported an Equal Error Rate (EER) of 7.5%.

Saevancee and Bhattarakosol [5] reported an EER of 1% using the K-Nearest Neighbor (KNN) algorithm and reported similar results using neural networks [6]. However, they conducted their experiments only using a notebook touchpad. A recent study conducted by Saira et al. [7], on smartphones, revealed that the keystroke pressure might not be unique and reported an EER of 8.4% when used in conjunction with classical keystroke features (timings).

2.2 Sensor-Assisted Keystroke-Based User Authentication

Recent literature reports the feasibility of using sensor data in combination with keystrokes for user authentication.

Several projects have been conducted to study the use of accelerometers and gyroscopes. For example, Giuffrida et al. [3] introduced *UNAGI*, a *fixed-text* and sensor-enhanced authentication mechanism for Android phones. They evaluated their method with 20 subjects and achieved an EER of 4.97% for passwords, and 0.08% for only sensor data. Miluzzo et al. [8] used sensor data to infer the icon activated by the user of iOS devices and reported 90% accuracy.

Similarly, Aviv et al. [9] present a method that relies on accelerometer data and keystroke timings to infer 4-digit PINs for unlocking smartphones. Specifically, they demonstrated the use of accelerometer data for learning user tapping and gesture-based inputs as these methods are required to unlock smartphones using PIN/password and graphical password patterns. Additionally, they collected data in two situations, *sitting* and *walking*.

Touchstroke is different from the previous solutions in terms of features (for sensors), classification strategies, number of sensors, sensor-data-acquisition and constraints on the input.

3 Background

3.1 Considered Sensors and Classifiers

Modern smartphones are equipped with multiple sensors with the capability to detect and compute device/user movement. Accelerometer and orientation

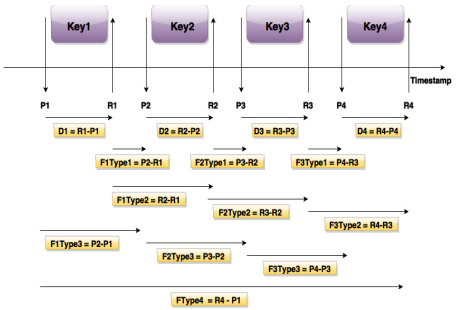


Fig. 1. Touchstroke features used in this paper, adapted from [2].

sensors are the most used sensors for movement recognition. Our solution uses seven three-dimensional sensors, for: three variants of the accelerometer; the gravity; the magnetic field or magnetometer; the gyroscope; and the orientation. All these physical sensors generate a continuous stream in 3 dimensions. We also added a fourth dimension to all of these sensors and name it magnitude, e.g. this dimension for the accelerometer is calculated as follows:

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

where a_x, a_y and a_z are the readings from the accelerometer sensor along the X, Y, Z dimensions, respectively.

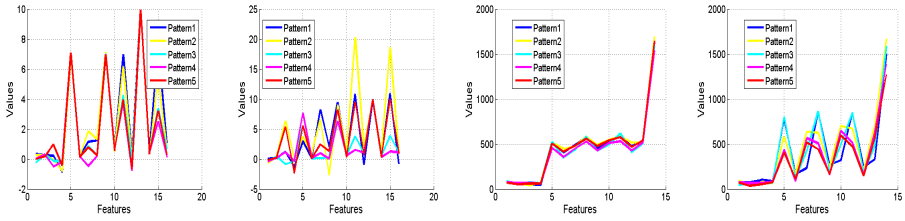
Classification is a way of comparing an unknown query input sample with the stored templates. Classifier selection depends on type and size of the dataset. We selected two classifiers by considering their short computation time and their resistance against over-fitting. Normally, Bayesian classifiers work well on small datasets and a random forest classifier is equally good for small and large datasets. We have applied these classifiers (with default parameters) in portable GUI-based Weka Experimenter Workbench.

3.2 Performance Metric

- True Acceptance Rate (TAR): The fraction of correctly accepted attempts of a real user.
- False Acceptance Rate (FAR: The fraction of incorrectly accepted attempts of an adversary.
- False Rejection Rate (FRR): The fraction of incorrectly rejected real users.
- Receiver Operating Characteristics (ROC): A parameter for the measurement of classifier performance. Specifically, a graphical representation of FAR vs FRR.

4 Intuition Assessment

It is our intuition that each user has a different way of holding and moving the phone when entering his PIN/password. If a user is holding a phone in his



(a) User1 (b) User2 (c) User1 (d) User2

Fig. 2. Comparison of 5 patterns of accelerometer (a and b) and touchstroke data (c and d), in *sitting* position for two users.

hand, it is very challenging for others to generate exactly the same movement pattern. Even in case of a successful mimicry, the movement pattern will still be different due to the differences in the structure of the human body (e.g., the height and exact orientation of the phone). Physical sensors can compute these minute differences. Similarly every user has a unique way of inputting data on a smartphone. An adversary can spoof and copy what is being written but it is very difficult to copy the exact timings of touch-types. Our assumption holds true if the patterns of the same user are similar enough and different enough from other users.

We argue (see Figure 2) that the patterns of the same user are very similar to each other and patterns of two users are different enough. Owing to space limitations, we show the patterns of raw accelerometer and touchstroke sensors for a single situation: when the user is sitting.

5 Experimental Evaluations

In order to validate our initial intuition we ran a set of experiments, described in the sections below.

5.1 Data Collection

We implemented *Touchstroke* as an Android application that triggers all physical sensors from the first touch-type and stops them after the last touch-type. At this moment the app is designed for only four touch-types with the possibility to be extended. We recruited 12 volunteers for our experiment; most of them are either MSc or PhD students but not security experts. In order to check the usability of our proposed mechanism, we collected data in six different user positions, i.e., *sitting*, *standing*, *walking*, *lying on sofa*, *walking upstairs* and *walking downstairs*. We used Google Nexus 5 running KitKat 4.4.2 for data collection. We collected 30 patterns from each user in each activity. In total, we collected 180 samples (in all 6 activities) from each user, making a total of 2160 samples from 12 users.

After registering the sensor with registerlistener(), data can be collected in both fixed and customized user-defined intervals. Android supports four fixed delivery rates, termed Sensor Delay Modes, namely, SENSOR_DELAY_FASTEST with a fixed delay of 0 sec, SENSOR_DELAY_GAME with a fixed delay of 0.02 sec, SENSOR_DELAY_UI with a fixed delay of 0.06 sec and the last one,
SENSOR_DELAY_NORMAL with a fixed delay of 0.2 seconds.

Touchstroke collects sensor data in SENSOR_DELAY_GAME mode.

5.2 Feature Extraction

We have four data streams from every three-dimensional sensor. We chose statistical features because it is computationally cheaper to compute them. We extracted 4 statistical features, namely mean, standard deviation, skewness and kurtosis from each data stream. In this way data from every sensor is transformed into a 4 by 4 feature matrix. Thus we have 16 features from all four dimensions of each sensor. Similarly, we extracted 14 features (see Figure 1), based on touch-typing timing, from the *text-independent* 4-digit PIN/password entered by the user.

Table 1. BayesNET classifier results for fused data for *standing*, *walking*, *lying on sofa*, *walking upstairs* and *walking downstairs*(averaged over all 12 users).

	Standing			Sofa			Walking			Upstairs			Downstairs		
Sensors	TAR	FRR	FAR	TAR	FRR	FAR	TAR	FRR	FAR	TAR	FRR	FAR	TAR	FRR	FAR
Raw + Touch	0.98	0.02	0.02	0.98	0.02	0.02	0.99	0.01	0.01	0.97	0.03	0.03	0.98	0.02	0.03
LPF + Touch	0.98	0.02	0.02	0.98	0.02	0.01	0.99	0.01	0.01	0.97	0.03	0.03	0.97	0.03	0.03
HPF + Touch	0.97	0.03	0.03	0.96	0.04	0.04	0.97	0.03	0.03	0.96	0.04	0.04	0.96	0.04	0.04
Grav + Touch	0.98	0.02	0.02	0.98	0.02	0.02	0.98	0.02	0.02	0.97	0.03	0.03	0.97	0.03	0.03
Gyro+Touch	0.97	0.03	0.03	0.96	0.04	0.04	0.98	0.02	0.02	0.96	0.04	0.04	0.97	0.03	0.03
Mag + Touch	0.97	0.01	0.01	0.99	0.01	0.01	0.96	0.04	0.04	0.95	0.05	0.05	0.96	0.04	0.04
Orient + Touch	0.98	0.02	0.02	0.97	0.03	0.03	0.98	0.02	0.02	0.96	0.04	0.03	0.97	0.03	0.03

5.3 Data Fusion

Data fusion can be done at the sensor level, feature level, match score level, rank level and decision level. Data fusion at an early stage may be more productive. However, sensor level fusion is not the best choice because of the presence of noise during data acquisition. Since feature representation shows much more relevant information corresponding to the class, the fusion at feature level is expected to provide better results. Thus, we fused data at feature level, in order to provide maximum relevant information to our recognition system. We fused the feature vector of each sensor with the touch-type feature vector, making a feature vector of 30 features. The reason for fusing only two sensors is to prevent over-fitting. Larger feature vectors may end up with over-fitting of the classifier.

5.4 Analysis

We used Weka Experimenter Workbench for the classification of these patterns. Data files were converted to ARFF files and later these ARFF files and two classifiers were added to the Weka Experimenter Workbench. We collected 30 observations for each activity from each user. We performed stratified cross-validation for training and testing of both classifiers for two reasons; firstly, because of equal number of patterns from each class assuming that it will arrange the data such that in each fold, each class comprises around half the instances. Secondly to test the classifiers with maximum possible user patterns. We present our results in terms of TAR, FRR, FAR and ROC curves.

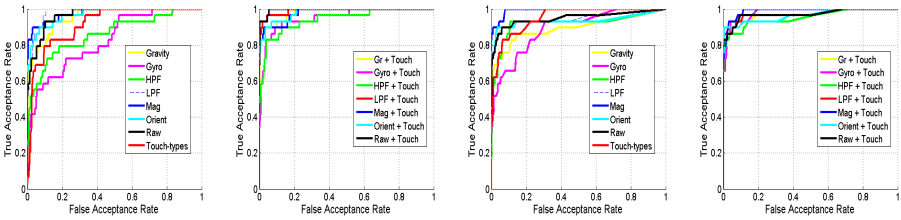


Fig. 3. ROC curve for BayesNET (a) for Individual and (b) for fused sensors and random forest (c) Individual and (d) fused sensors.

6 Discussion of Results

We achieved acceptable authentication rates for all the activities from individual sensors especially variants of accelerometers. As it can be very difficult to type while *walking*, *going downstairs* and *upstairs*, we can expect a little increase in error rates in those situations. However, *Touchstroke* performed well even in these positions, yielding acceptable authentication results (see Table 1).

The purpose of fusion of each sensor with touchstroke data is twofold. Firstly, to improve authentication accuracy; ROC curves for both the classifiers show an improvement in accuracy for fused data (see Figures 3b and 3d). Secondly, to make the system more secure; it is comparatively difficult to mimic two behaviors at the same time. Due to space limitations, we present ROC curves for *sitting* activity and authentication results of the BayesNET classifier (see Table 1) for fused data only.

Another important observation is related to the way users hold the phone. Some users use one hand and others use both hands for holding and entering the *text-independent* text. *Touchstroke* works for both types of user. Our experiments are preliminary since we run the tests with a limited number of users who are not representative of the general population, thus we cannot exclude some bias due to the particular composition of our test set.

7 Conclusions and Future Work

We propose a bi-modal biometric system, *Touchstroke*, for smartphone user authentication based on phone movement patterns and *free-text* 4-digit touch-type patterns. We implemented and evaluated the system on Android smartphones. The initial experiments indicate that our solution is highly accurate in each situation. Each sensor can potentially be used with touch-type features for user authentication. Our solution can be implemented in any off-the-shelf smartphone without the need for additional hardware, hence can be used as a stand-alone method or can be complemented by traditional passwords for additional security.

In future work, we will test whether or not the fusion of multiple sensors and/or with touchstrokes has an impact on accuracy. Futher, in order to check the impact of the length of the touch-type, we will investigate whether or not typing a long-digit password/PIN gives different results from those obtained for 4-digit entries.

References

1. Raza, M., Iqbal, M., Sharif, M., Haider, W.: A survey of password attacks and comparative analysis on methods for secure authentication. World Applied Sciences Journal **19**, 439–444 (2012)
2. Teh, P.S, Teoh, A.B.J., Yue, S.: A survey of keystroke dynamics biometrics. The Scientific World Journal, Hindawi Publishing Corporation (2013)
3. Giuffrida, C., Majdanik, K., Conti, M., Bos, H.: I sensed it was you: authenticating mobile users with sensor-enhanced keystroke dynamics. In: Dietrich, S. (ed.) DIMVA 2014. LNCS, vol. 8550, pp. 92–111. Springer, Heidelberg (2014)
4. Huang, X., Lund, G., Sapeluk, A.: Development of a typing behavior recognition mechanism on android. In: Proceeding of the IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 1342–1347. IEEE, Bradford (2012)
5. Saevanee, H., Bhatarakosol, P.: User authentication using combination of behavioral biometrics over the touchpad acting like touch screen of mobile device. In: Proceeding of the International Conference on Computer and Electrical Engineering (ICCEE 2008), pp. 82–86. IEEE, Phuket (2008)
6. Saevanee, H., Bhatarakosol, P.: Authenticating user using keystroke dynamics and finger pressure, In: Proceedings of the 6th IEEE Consumer Communications and Networking Conference (CCNC 2009), pp. 1–2. IEEE, Las Vegas (2009)
7. 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)
8. Miluzzo, E., Varshavsky, A., Balakrishnan, S., Choudhury, R.R.: Tapprints: your finger taps have fingerprints, In: Proceedings of the 10th international conference on Mobile systems, applications, and services, pp. 323–336. ACM (2012)
9. Aviv, A.J., Sapp, B., Blaze, M., Smith, J.M.: Practicality of accelerometer side channels on smartphones, In: Proceedings of the 28th Annual Computer Security Applications Conference, pp. 41–50. ACM (2012)