SnapAuth: A Gesture-Based Unobtrusive Smartwatch User Authentication Scheme

Attaullah Buriro[1,2](✉), Bruno Crispo[1,3], Mojtaba Eskandri[4], Sandeep Gupta[1], Athar Mahboob[2], and Rutger Van Acker[5]

[1] Department of Information Engineering and Computer Science (DISI), University of Trento, Trento, Italy
{attaullah.buriro,bruno.crispo,sandeep.gupta}@unitn.it

[2] Department of Information Security, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan, Pakistan

[3] Department of Computer Science, DistriNET, KU Lueven, Leuven, Belgium

[4] Create-NET, Foundazione Bruno Kessler (FBK), Trento, Italy

[5] Swift via Exellys, Hamont-Achel Hamont, Flanders, Belgium

Abstract. In this paper, we present a novel motion-based behavioral biometric-based user authentication solution - SnapAuth, for Android-based smartwatch. SnapAuth requires the user to perform finger-snapping (Snapping (or clicking) one's fingers is the act of creating a snapping or clicking sound with one's fingers. Primarily this is done by building tension between the thumb and another (middle, index, or ring) finger and then moving the other finger forcefully downward so it hits the palm of the same hand at a high speed [4].) action, while wearing the smartwatch to perform the authentication. SnapAuth profiles the arm-movements by collecting data from smartwatch's built-in accelerometer and gyroscope sensors, while the user performs this action. We implemented and evaluated SnapAuth on Motorola Moto 3G smartwatch. SnapAuth could be widely accepted by users as it utilizes the users' familiarity with the very common finger-snapping action and users do not need to remember any secret.

Keywords: Security · Authentication and access control
Behavioral biometrics · Smartwatch

1 Introduction

The use of smartwatches is steadily and constantly increasing in recent years. Since, they are typically a personal device that users hold all the time, they are an obvious candidate device to support authentication of its owner.

Authentication is the process of restricting the device access to the legitimate users, only.

Classical authentication schemes, such as PIN/password establish the identity with what the user remembers. These solutions are neither considered secure [1] nor usable [2,3], because they require users to remember their secret

A. Saracino and P. Mori (Eds.): ETAA 2018, LNCS 11263, pp. 30–37, 2018.
https://doi.org/10.1007/978-3-030-04372-8_3

and enter it every time they need to use the device. Additionally, entering text or sketching a graphical password on smartwatches could be extremely difficult because of the small size of the touchscreen.

Biometric-based authentication establish the identity through biological modalities, such as face, fingerprint, retina, etc. These solutions have already been implemented on recent smartphones, e.g., Apple Face ID [5] and fingerprint sensors in iPhones [6], however, these schemes, being non-transparent, have still the main disadvantage of annoying the user [7]. Further, the data of some of these modalities can be stolen as easily as passwords [10].

Behavioral biometric, e.g., swiping, touch-dynamics, etc., seems a better option for the development of user authentication schemes for the new generation of personal devices, because they are dependent on the person-specific user actions and habits, which makes them more attractive towards implicit/unobtrusive user authentication [11].

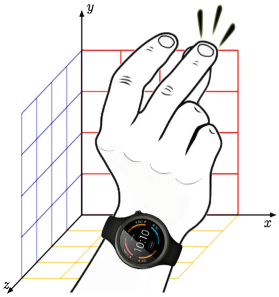


Fig. 1. Finger-snapping in 3d space

In this work, we present a motion-based finger-snapping gesture-based user authentication scheme - SnapAuth, for smartwatch unlocking. The scheme collects the arm movements fingerprints through the accelerometer and gyroscope sensors in three dimensions, while the user performs the finger-snapping gesture (as depicted in Fig. 1), and performs user profiling. More specifically SnapAuth collects the arm-movement generated data, from accelerometer and gyroscope, for the short duration, during finger-snapping gesture, at a sample rate of 50 samples/s and performs the identity confirmation. SnapAuth, using simple, yet effective, state-of-the-art machine learning classifiers, decides if the smartwatch is worn by the legitimate user or by an impostor. Access to the smartwatch is granted in case the user is confirmed as the legitimate user otherwise its denied. SnapAuth neither requires any token or password from the user, thus, the scheme is completely unobtrusive, and usable for smartwatch unlocking. We validated SnapAuth on a real device (Motorola Moto 3G) and obtained promising results.

The main contributions of this paper are:

- The proposal of SnapAuth - an arm-motion-based user authentication scheme for Android smartwatch. The scheme authenticates the users based on the differences in the arm-movements generated while user performs finger-snapping action.
- Proof-of-the-concept prototype implementation of the scheme on a smartwatch.

2 Related Work

Behavioral biometric-based user authentication using smartwatches has been already explored by few papers. Draw-a-pin [12] leverages the user behaviour while drawing a PIN and the correctness of the PIN, to authenticate the user. Authors achieved 20.36% average error rate on their collected dataset of 30 participants, in two activities, i.e., sitting, walking, in lab settings using Samsung Gear Live smartwatch.

Lewis et al. [13] proposed a motion-based authentication solution for smartwatch users. The system exploits the free-form arm movement as a behavioral biometric modality for user authentication. By applying DTW as a classifier on their collected dataset of 5 users, authors achieved the accuracy up to 84.6%. Similarly, the other relevant study "VeriNET", takes motion signals as password, and uses the deep recurrent neural network to authenticate the users [15]. Authors evaluated their scheme on 310 participants on $\approx 60k$ passcode entries and achieved an Equal Error Rate (EER) of 7.17% on PINs and 6.09% on Android lock patterns.

We consider the study by Kumar et al. [14] very relevant to our work. Authors proposed four variants of continuous user authentication design based on users arm movements while walking. The design incorporated smartwatchs accelerometer and gyroscope sensor data, individually as first and second variants, and then, applied feature and score-level fusion as the third and fourth variant. The system was tested under 3 different environments, i.e., intra-session (40 users dataset), inter-session (40 users dataset), and inter-phase (12 users dataset) using 4 classifiers, namely, k nearest neighbors (k-NN) with Euclidean distance, Logistic Regression, Multilayer Perceptrons, and Random Forest resulting in a total of sixteen authentication mechanisms. They achieved the mean dynamic False Accept Rate (DFAR) of 0% and Dynamic False Reject Rate (DFRR) of 0% for all of the twelve authentication mechanisms in the intra-session environment. In the inter-session environment, k-NN performed best with a mean DFAR of 2.2% and DFRR of 4.2% for a feature level fusion-based design. Whereas, in the inter-phase environment, the DFAR and DFRR increased to 15.03% and 14.62% respectively for the same feature level fusion-based design with the k-NN classifier.

SnapAuth is different from existing state-of-the-art authentication solutions in the following ways: (i) it leverages a novel finger-snapping action that is easy to perform, and (ii) the data collection is fully unobtrusive making it suitable for designing frictionless user authentication solutions.

3 Methodology

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

3.1 Considered Hardware

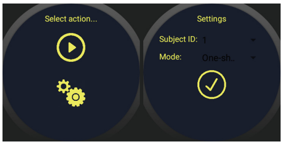
This work uses the Motorola Moto 360 smartwatch for implementation of the proposed authentication scheme. For both data collection and the validation phase, the considered sensors offered by the smartwatch are used for the generation of raw sensor data.

3.2 Smartwatch Sensors

Android categorizes built-in sensors in three types, i.e. motion, environmental and positional, in their API guide on sensors[1]. To capture gestures, applications in this work uses two built-in motion sensors available on the Moto 360 to measure the acceleration, and rotation during the performance of a gesture. Moto 360 delivers a 50 Hz sampling rate for both the accelerometer and gyroscope meaning that both sensors are able to ideally generate 50 samples per second with an error margin around ±3 samples per second.

3.3 Data Collection

We developed a customized Android application, namely *SnapCollector* to collect the finger-snapping gesture. *SnapCollector* can be installed on any Android-based smartwatch having Android version 4.4 or higher, installed. We collected accelerometer and gyroscope readings at highest sample rate (50 Hz) because this sample rate was found empirically suitable for authentication purposes, in recent studies [18]. Figure 2 shows the main screen of our developed application.



Subject ID: 1
Mode: One-sh..

Fig. 2. Main screen and settings

We recruited 11 volunteers (8 males) to participate in our three-day long three-session experiment. The participants had a background in computer science. We asked them about the natural hand and in which wrist they usually wear their watch.

We collected data from an experiment spanned over three-sessions. We ensured that all the recruited participants had to participate in these sessions on three consecutive days. The motivation was to check the performance of the gesture in intra-session and inter-session analysis (Fig. 3).

[1] https://developer.android.com/guide/topics/sensors/sensors_overview.html.

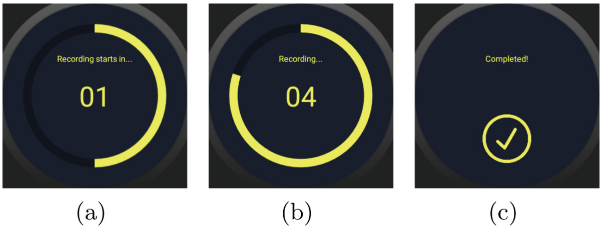


Fig. 3. (a) start of data recording & (b) data recording (c) recording complete.

3.4 Feature Extraction

The collected raw data from both accelerometer and gyroscope sensors is three dimensional, i.e., streams in X, Y, and Z dimensions. We also computed another dimension using the following equation and termed it as Magnitude.

$$S_m = \sqrt{s_x^2 + s_y^2 + s_z^2}$$

where S_m represents the magnitude of sensor S and s_x, s_y and s_z represent the values of the X, Y and Z stream respectively from sensor S. We extracted four statistical features, namely, mean(μ), standard deviation(σ), skewness(γ) and kurtosis(γ'), from every acquired raw stream and concatenate them to form a feature vector.

3.5 One-Class Classifier Selection

We chose three simple, yet effective state-of-the-art machine learning classifiers, namely, Bayes NET (BN); Multilayer Perceptron (MLP); and Random Forest (RF) to perform the classification. We chose these classifiers because they were found extremely accurate in the previous studies [16–18]. We used Weka workbench and used meta-class classifier - the *OneClassClassifier*[2], for our analysis.

4 Analysis

We collected 10 observations per activity per user (in total 50 observations in 5 activities). In first iteration, we picked the first two observations from each activity (just 10, in total) and trained the chosen classifiers on those observations. Remaining 40 observations were used as the testing set to perform the classification. In the second iteration, we picked 3 observations from each activity (15, in total) and trained the classifier on them, and the remaining 35 observations were used to test the classifier. We used max 15 observations for training, for two reasons: firstly because its common to get less number of training samples from

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

users in real world, i.e., signature samples in banks, and secondly the users are reluctant to provide more training samples and might get annoyed if the system requires more training samples.

User authentication on smart devices is essentially a one-class classification problem where the data from one user - "the owner" is used to train the classifier and later that classifier is tested on the remaining samples of that one user - to obtain True Accept Rate (TAR) and False Reject Rate (FRR), and on the data of all the would-be adversaries (to obtain False Accept Rate (FAR) and True Reject Rate (TRR) [17]. We followed the above-mentioned scheme and trained all the chosen classifiers on the data of one user and tested in two settings (as mentioned above). The process is repeated for all the 11 users and the obtained average results are reported. Figure 4 drawn in the KnowledgeFlow module of Weka to perform both training and verification of One-class classifiers.

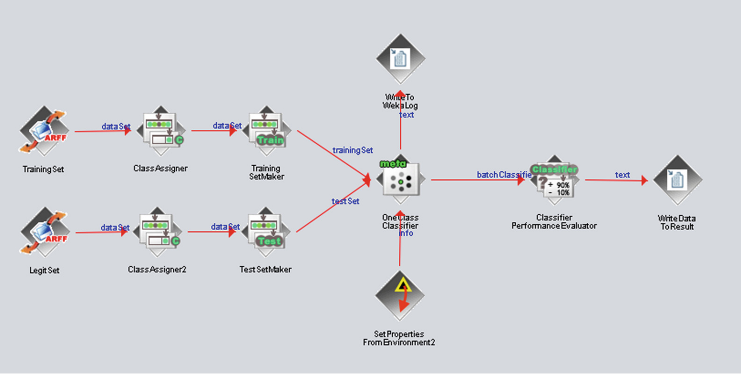
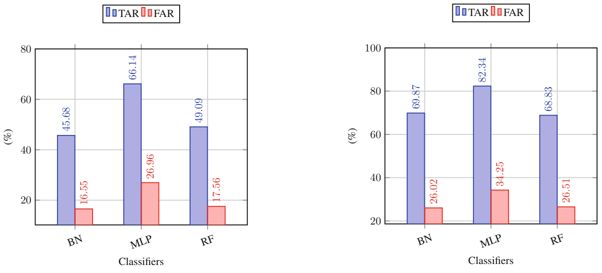


Fig. 4. Authentication scheme

5 Results

The accuracy of any biometric-based authentication system is normally reported in terms of TAR, FRR, FAR, or EER. We are reporting our obtained results in terms of TAR and FAR only as $TAR = 1 - FRR$ and $FAR = 1 - TRR$. Figure 5 depicts our obtained results (with default settings of all the classifiers) on full features.

MLP classifier performed comparatively better (see Fig. 5) as compared to its counterparts, i.e., BN and RF. We obtained 66.14% TAR at ≈27% FAR with default settings on just 10 training samples. TAR further improved with the increase in the number of training samples, i.e., training on 15 samples, provided 82.34% TAR, however, the FAR also increased (34.25%). The reason behind the increase of FAR is the less number of training samples and this could



(a)

(b)

Fig. 5. Results of all the classifiers on 10 (a) & 15 (b) training samples.

further be improved if the training is performed over the significant number of training samples, i.e., 25.

SnapAuth is clearly in initial stages and thus a bit less accurate (but enough to prove the initial intuition that fingersnaps can be used to authenticate users). We expect its accuracy to be improved using more testers and fine tuning the classifier.

6 Conclusions

This work proposes a simple, unobtrusive, and novel motion-based smartwatch user authentication scheme. The scheme exploits the very common human behavior for user authentication purposes. It authenticates users based on the differences in the arms' micro-movements, collected through smartwatch sensors, while the user performs the finger-snapping action. SnapAuth is user-friendly and easy for the users because they do not require any secret to remember and/or to type. The scheme leverages the built-in hardware, so it does not require any additional dedicated hardware and hence avoids additional costs.

As future work, we are planning to repeat the experiments in the wild, possibly using a crowd-sourcing platform, thus increasing considerably the number of testers. We will also investigate and report the performance and usability of our scheme by performing usability related experiments. We will also perform accurate tests about the security of the scheme and how easy is for an attacker to spoof or mimic the behavior of legitimate users.

References

1. Lashkari, A.H., Farmand, S., Zakaria, D., Bin, O., Saleh, D.: Shoulder surfing attack in graphical password authentication. arXiv preprint: arXiv:0912.0951 (2009)
2. Davis, D., Monrose, F., Reiter, M.K.: On user choice in graphical password schemes. In: USENIX Security Symposium, vol. 13 (2004)

3. Gupta, S., Buriro, A., Crispo, B.: Demystifying authentication concepts in smartphones: ways and types to secure access. Mob. Inf. Syst. (Hindawi) **2018**, 16 (2018)
4. Finger snapping. https://en.wikipedia.org/wiki/Finger_snapping. Accessed 20 June 2018
5. About Face ID advanced technology. https://support.apple.com/en-us/HT208108. Accessed 20 June 2018
6. How the iPhone 5S Fingerprint Scanner Works and What It Means For You. https://gizmodo.com/how-the-iphone-5ss-fingerprint-scanner-works-and-what-1265703794. Accessed 20 June 2018
7. De Luca, A., Hang, A., Von Zezschwitz, E., Hussmann, H.: I feel like I'm taking selfies all day! Towards understanding biometric authentication on smartphones. In: 33^{rd} Annual ACM Conference on Human Factors in Computing Systems, pp. 1411–1414 (2010)
8. Windows Hello face recognition spoofed with photographs. https://nakedsecurity.sophos.com/2018/01/02/windows-hello-face-recognition-spoofed-with-photographs/. Accessed 20 June 2018
9. iPhone 6 vulnerable to TouchID fingerprint hack. https://www.itnews.com.au/news/iphone-6-vulnerable-to-touchid-fingerprint-hack-392414. Accessed 20 June 2018
10. Hacker fakes German minister's fingerprints using photos of her hands. https://www.theguardian.com/technology/2014/dec/30/hacker-fakes-german-ministers-fingerprints-using-photos-of-her-hands. Accessed 20 June 2018
11. Buriro, A.: Behavioral biometrics for smartphone user authentication. Ph.D dissertation. University of Trento, Italy (2017)
12. Nguyen, T., Memon, N.: Smartwatches locking methods: a comparative study. In: Symposium on Usable Privacy and Security (SOUPS) (2017)
13. Lewis, A., Li, Y., Xie, M.: Real time motion-based authentication for smartwatch. In: IEEE Conference on Communications and Network Security (CNS), pp. 380–381 (2016)
14. Kumar, R., Phoha, V.V., Raina, R.: Authenticating users through their arm movement patterns. arXiv preprint arXiv:1603.02211 (2016)
15. Lu, C.X., Du, B., Kan, X., Wen, H., Markham, A., Trigoni, N.: VeriNet: user verification on smartwatches via behavior biometrics. In: Proceedings of the First ACM Workshop on Mobile Crowd sensing Systems and Applications, pp. 68–73 (2017)
16. Sitova, Z., et al.: HMOG: A New Biometric Modality for Continuous Authentication of Smartphone Users. arXiv preprint arXiv:1501.01199 (2015)
17. Buriro, A., Akhtar, Z., Crispo, B., Gupta, S.: Mobile biometrics: towards a comprehensive evaluation methodology. In: IEEE International Carnahan Conference on Security Technology (ICCST), pp. 1–6 (2017)
18. Buriro, A., Crispo, B., Delfrari, F., Wrona, K.: Hold and sign: a novel behavioral biometrics for smartphone user authentication. In: Proceedings of the IEEE Security and Privacy Workshops (SPW), pp. 276–285 (2016)