Demo of HeadBanger: Authenticating Smart Wearable Devices Using Unique Head Movement Patterns

Sugang Li*, Ashwin Ashok†, Yanyong Zhang*, Chenren Xu‡, Macro Gruteser*, Janne Lindqvist*
*WINLAB, Rutgers University, North Brunswick, NJ, USA
†Carnegie Mellon University, Pittsburgh, PA, USA
‡CECA, Peking University, Beijing, China

***Abstract*—We demonstrate a system for direct authentication of users to their head–worn wearable device through a novel approach that identifies users based on motion signatures extracted from their head–movements. This approach is in contrast to existing indirect authentication solutions via smartphone or using touch–pad swipe patterns. The system, dubbed *Headbanger*, is a software authentication solution that leverages unique motion patterns created when users shake their head in response to music played on the head–worn device, and sensed through integrated accelerometers. In this demo, we demonstrate *Headbanger* on Google GLASS and show the effectiveness of the system in two authentication modes, that include (i) a trained user reliably authenticated to the owned GLASS device, and (ii) an attacker being prevented from login when attempting to login to the GLASS device by imitating the owner's head–movements.**

I. Introduction

Wearable devices are increasingly becoming an integral part of the pervasive sensing and computing infrastructure today. These devices often collect data about users, which are also frequently shared to the users' multiple devices; for example, as background notifications. In such a wireless network of human sensing wearable devices the concerns of owner's privacy and security from malicious users are paramount, thus calling for an effective and robust user–authentication solution for wearable devices. Prior works have explored this problem in point–specific applications for limiting privacy threats [7]–[9], for example from photography–enabled wearable devices. In addition, existing approaches for authentication on wearables are limited to an indirect method – channeling through a smartphone [4] or using swipe patterns integrated touch–pads [3]. Such approaches impose a requirement for the user to carry both device, smartphone and the wearable, all the time, and also have an increased chance of being compromised if either of the devices are lost or stolen. Therefore, a direct, secure and robust authentication solution for wearable devices is the need of the day.

In this work, we propose a novel approach for direct authentication on wearable devices through signatures generated from the device user's motion patterns. In particular, we design a system, *Headbanger*, for head–worn wearable devices, that identifies users based on their head–movement patterns. We design a software system, in the form of an authentication service on a head–worn wearable device, that authenticates users to their device by recognizing their unique head–movement signatures. This approach is in contrast and more secure than linking the wearable device to user's email [3] and other online accounts [1]. In this work, we demonstrate *Headbanger*through an authentication service implementation on Google GLASS and show its reliability to user–authentication and robustness to imitation attacks from adversaries. In Fig. 1 we illustrate the two modes of authentication, using *Headbanger*, that we have implemented and will demonstrate.

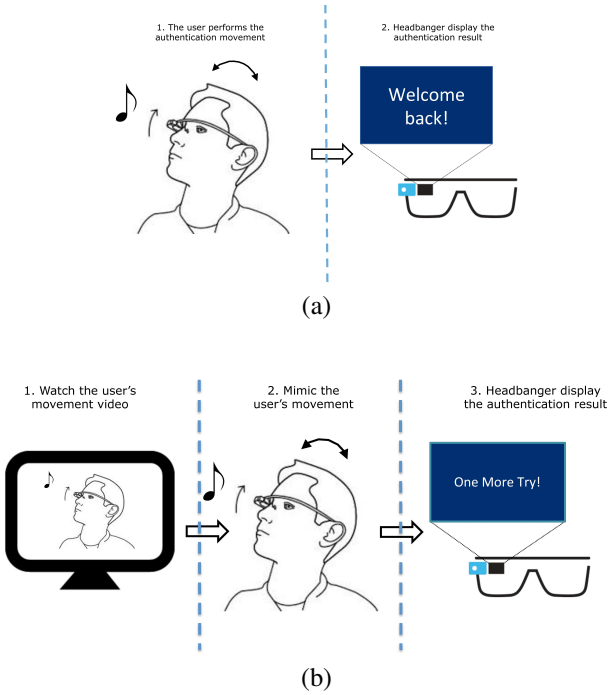

Fig. 1. Illustration of Headbanger demo when (a) authenticating owner for login, (b) preventing an attacker from login.

In the following sections we will describe the *Headbanger*system design briefly in Section II, followed by a description of the implementation and demonstration plans in Section III.

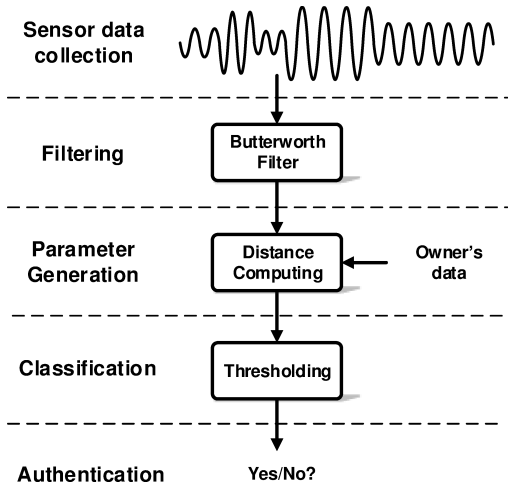

Fig. 2. *Headbanger* system design flow. The online authentication phase of *Headbanger* consists of the following steps: (1) sensor data collection in which we collect accelerometer data while users move their head as a response to an audio track played on the glass, (2) filtering in which we apply a Butterworth filtering to smoothen the sensor data for subsequent processing, (3) parameter generation in which we calculate the distances between two accelerometer samples as the parameter, and (4) classification in which we adopt an adaptive thresholding mechanism to classify the user's head movement, whose output will be used as the authentication result.

II. System Design

We design the *Headbanger* system as a software service that provides direct authentication for users through their head–movements, to login to their smart–glass devices and authenticating to apps running on the device. We envision that *Headbanger* will execute upon device power–up or wake–up or application initiation; a mechanism that is similar to that of screen unlocking in smartphones. In *Headbanger*the authentication process is divided into two steps: a training phase and an online authentication phase. In the training phase, the system collects accelerometer data corresponding to the user's head movements stimulated through music cues for a specific duration, and uses the data samples to build a binary classifier. For simplicity, in our current implementation we execute the training–phase off–line. In reality, the training data would be accumulated over time to keep the system up-to-date. Extending the implementation towards on–line training will only require minor modifications to the software modules in our current implementation. In the online authentication phase, the user performs the same head movement (as in the training phase) under the same music cue settings and attempt to authenticate to the device. *Headbanger* modules collect the samples and labels them based on the local training data–base and the online classifier output. The user is authenticated to the device if the output from the binary classifier is *successful*, and not *failed*.

As shown as Fig. 2, the authentication process in *Headbanger* involves four key components: sensing, filtering, distance computing and classification. We briefly discuss these system design components in the following subsections.

A. Sensing and Filtering

The sensing step involves collecting data samples from the built–in accelerometer when the user performs head–movements, in response to the given music cue with duration of T seconds. The raw sensor data is collected from 3 axis at a sampling rate of r points/sec. Thus, each sample is a matrix of $3 \times rT$ points. T is set to the order of few seconds, considering the fact that frequency of human movements are in that order. A low–pass Butterworth filter [6], using a cut–off frequency of 10 Hz, then removes high frequency noise from the sensor raw data.

B. Distance Computing

This step is the preparation stage for a binary classifier. We adopt a simple technique for distance computation, through Dynamic Time Warping (DTW) [5], as it is feasible to implement the same even on computing resource–constrained wearable device. We use the DTW distance in our implementation. Let us suppose that there are two time series samples $S_a = (s_1, s_2, ..., s_n)$ and $S_b = (s_1, s_2, ..., s_n)$. The DTW distance is defined as the temporal separation between the samples when the optimal alignment between S_a and S_b has been determined by time–warping the two series in a non–linear fashion [5].

C. Classification

The classification phase determines a test sample as *successful* or *failed* based on a threshold. Thresholding–based classifier strikes a good balance between simplicity and good performance.

For the training phase, we design a technique for choosing $Top - K$ template samples from the whole training set. We compute an average distance for each sample by computing the DTW distance of each sample to all other samples. The top K samples with the smallest average distance are chosen as templates. These K samples represent the sample space, as they are essentially an empirical estimation of the centroid of the data sample space. In the classifier, only the $Top - K$ samples are contained thus reducing the computation load and adding robustness against randomness of the samples in the system. We establish a threshold for the each of the $Top - K$ samples. It is determined by average value of the distance, μ_s, and the standard deviation, σ_s. The threshold is defined as $\mu_s + n\sigma_s$, where n is a tunable parameter in the classifier, and can take real positive number values.

In the online authentication phase, if the DTW distance between the test sample and the template is below the threshold for the chosen template, we label the test sample as *successful*, otherwise as *failed*. Since an classification output if obtained for each of the $Top - k$ samples, we chose the sample based on a voting strategy that choses based on majority. Eventually, if the user's test data is classified as *successful* the user is accepted (authenticated for login) by the system; otherwise, the user is rejected.

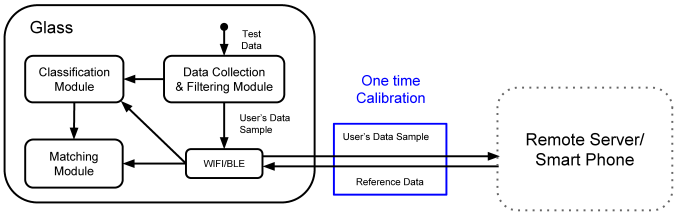

Fig. 3. Software modules of *Headbanger* implementation

III. Demonstration

A. Headbanger App on GLASS

We implemented *Headbanger* as an app on Google Glass using the Glass Development Kit (GDK) [2]. Fig. 3 shows the application components in our implementation. In the initiation phase, the application plays a music cue for a user–specified duration. The user performs head movement by following the music flow while the application is collects accelerometer data. On completion of the music cue, the application enters the processing phase where the sensor data is input to the *Headbanger* 's software modules for processing. In our current implementation, the computation for the training phase is done on a PC and the template is determined apiori to the online authentication process. As a result, the app outputs a binary decision which is shown as a textual output on the GLASS's screen; *successful* is referred as "Welcome back!", and *failed* is referred as "One more try!", as shown in Fig. 1.

B. Demonstration Setup

We will demonstrate *Headbanger*in two authentication modes for login to GLASS : (i) owner authenticating to the device, and (ii) attacker being prevented from login to the device. For the former, the demo presenter, considered as one of the owners of the GLASS device, will demonstrate a correct authentication movement in order to show the capability of *Headbanger* to correctly recognize a true user. To demonstrate the robustness of *Headbanger*to attacks, the demo presenter will set up a 'mimic challenge'during the demonstration session. The device will be pre–loaded with template samples from 4 voluntary users (each is a potential attacker to others), including the presenter. All of users will have access to viewing a video recording of their head–movement exercise, while they are attempting to use the system, and then given a chance to login to the GLASS.

At the demo, the attackers will then be allowed to choose one of the users' action to mimic. In this regard, they will be allowed to watch the corresponding video for as long as they feel comfortable to start imitating the system. The demo app will provide a feedback on the similarity of their head–movements to the original, after each attempt, so that the challengers may adjust their actions if necessary. If the attacker fails to login within 10 attempts, the attack is considered to being compromised by our system. Our design uses the fact that, since the music cue is played via a bone–conduction speaker or a earplug, it is difficult to record the music in a normal environment, making it difficult for the attackers to break the system. To ensure fairness during the demo session, we will set the volume on the GLASS to maximum and try to achieve audible quality recording of the music cue played on the GLASS.

During the entire session at the demo, we will screen–cast the Google GLASS's heads–up display screen to an external monitor for audience view and participation to emulate a real–time testing environment of our system.

IV. Conclusion

As wearable devices are gradually integrating into human lives, providing security and privacy protection for these devices becomes critically significant. We have developed a user authentication system, *Headbanger*, for directly authenticating a user to their head–worn device. Compared to the existing approaches, the proposed system provides a lightweight and convenient software solution without additional hardware. In this work, we implement and demonstrate the system as a proof–of–concept of direct authentication on head–worn wearable devices through head–movements. We show the reliability and robustness of the system to imitation attacks. In future, we aim to test the system at a larger scale with a goal to integrate head–movement based authentication as a fundamental service in wearable devices.

References

[1] Fitbit. http://en.wikipedia.org/wiki/Fitbit.
[2] Glass development kit. https://developers.google.com/glass/develop/gdk.
[3] Google glass. http://en.wikipedia.org/wiki/Google_Glass.
[4] Pebble. https://www.pebble.com.
[5] D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *ACM KDD workshop*, 1994.
[6] R. Challis and R. Kitney. The design of digital filters for biomedical signal processing part 3: The design of butterworth and chebychev filters. *Journal of biomedical engineering*, 1983.
[7] R. Hoyle, R. Templeman, D. Anthony, D. Crandall, and A. Kapadia. Sensitive lifelogs: A privacy analysis of photos from wearable cameras. 2015.
[8] R. Hoyle, R. Templeman, S. Armes, D. Anthony, D. Crandall, and A. Kapadia. Privacy behaviors of lifeloggers using wearable cameras. In *ACM UbiComp*, 2014.
[9] S. Jana, A. Narayanan, and V. Shmatikov. A scanner darkly: Protecting user privacy from perceptual applications. In *IEEE SP*, 2013.