Open source tools and hackable biofeedback devices

Vera Onunda  
Institute of Computer Science  
University of Tartu  
Tartu, Estonia  
onunda@ut.ee

Abstract—The study aims to investigate the possible use of open-source solutions to hack readily available biofeedback devices to bypass vendor-based software for further use in developing real-time biofeedback systems and research. Three smart bands that are cost-effective and readily available are selected and hacked using open-source software GadgetBridge for data extraction and management from devices’ biofeedback sensors. The study shows the benefits of using readily available devices and tools and how the data integrity of collected data from the sensors is not affected by comparing data displayed in the vendor application and data collected through the gadget bridge application.

Keywords—open-source, GadgetBridge, hackable, data biofeedback and sensors

I. INTRODUCTION

Hacking involves compromising digital devices or networks to gain unauthorized access to any account, computer system, or electronic device. Hacking in this context involves buying a readily available device with biosensors, installing open-source software for data collection, and using it for personal research. Open-source tools are free publicly accessible tools that can be modified and shared based on their design. Open-source tools are critical in helping solve the barriers that come with using biofeedback devices locked to vendor-specific software in research. Open-source software aims to: (1) Reduce the cost that comes with the use of vendor-based software in terms of pricing or customization to cater to research needs. (2) Offer flexibility in terms of research and development by allowing the development of new software products with improved functionality, processes, and efficacy. (3) It also allows for creativity and innovation to thrive in areas with research gaps, especially in developing real-time biofeedback systems.

The potential to fill in the research gaps seen in biofeedback research is enormous, especially with the support of the diverse open-source community. The technological advances and improvement of wearable biofeedback devices have created an opportunity for researchers to maximize readily available resources while enjoying the flexibility of support from the open-source community. New applications demand has led to advances in MEM (Micro-electromechanical systems) technologies [1] and better readily available biofeedback devices like smart bands. The recent formation of open-source communities eager to bypass vendor-based devices like GadgetBridge has created opportunities for researchers to capitalize on new technology and ways to bypass vendor-locked software linked to devices like smart bands, which had been a barrier to customization for research.

The research makes the following contributions:

- Show how hacked smart bands collect and manage real-time biofeedback data.
- Demonstrate hacked biofeedback use in the development of a multipurpose real-time biofeedback system.
- Compare the effectiveness of the hacked devices in monitoring body physiological functions.
- Show the benefits of using open-source tools and hacked biofeedback devices in research and development.

II. PROPOSED REAL-TIME BIOFEEDBACK SYSTEM

The following software and hardware components are selected proposed:

A. Hackable biofeedback devices for data collection include the following smart bands.
1. Amazfit band 5
2. Mi-band five
3. Mi-band two

<table>
<thead>
<tr>
<th>Table I.</th>
<th>HACKABLE BIOFEEDBACK DEVICES OVERVIEW (2021)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Features</td>
<td>Smart Bands</td>
</tr>
<tr>
<td></td>
<td>Amazfit band 5</td>
</tr>
<tr>
<td>Sensors</td>
<td>Triaxial acceleration sensor, PPG heart rate sensor, Capacitive wear sensor, PPG(Photoplethysmography)biotracking optical sensor, and 3-axis acceleration sensor or 3-axis gyroscope sensor</td>
</tr>
<tr>
<td>Health monitoring capabilities</td>
<td>Biotracker, sleep, heart rate monitoring, stress monitoring, blood-oxygen monitoring, SpO₂, women’s health</td>
</tr>
<tr>
<td>3rd-party app integration</td>
<td>GadgetBridge</td>
</tr>
<tr>
<td>Features</td>
<td>Amazfit Band 5</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Price Range</td>
<td>€ 45</td>
</tr>
<tr>
<td>Battery Life</td>
<td>15 days</td>
</tr>
</tbody>
</table>

Table I. Summary of selected smart bands and their capabilities.

In this research, the biosensors capabilities found in the three smart devices and uses include:

- The triaxial acceleration sensor monitors the vibrations from three directions or axes X, Y, Z in the orthogonal plane. It determines an object’s position in space and movement [2].

![Triaxial Acceleration Sensor](image1)

- PPG heart rate sensor monitors heart rate variability using infrared light that helps measure blood perfusion through tissues, blood oxygen saturation, and blood pressure. PPG sensors components include optical emitters, the two led lights that send light waves to the skin, a photodetector that captures refracted light from the user skin and saves data in binary format. Finally, the photodetector signal and the accelerometer combined calculate the PPG Algorithm and convert the binary data motion tolerant heart data. PPG Algorithms are also used to calculate calories, HRV (heart rate variability), blood oxygen level, and blood pressure [3]. PPG heart rate sensor takes into account the Beer Lambert’s Law that takes into account the linear relationship between absorbance, concentration, molar absorption coefficient, and optical co-efficient of a solution [4].

- A capacitive proximity sensor mainly detects if a user is wearing the device. It uses the electrical property of the capacitance and change of capacitance based on the oscillation around the active face of the sensor [5] [6]. In this, case the conductive target is the blood supply on the user’s epidermis. Capacitive proximity sensors generate an electrostatic field that changes the capacitance via an oscillatory circuit [6].

![PPG Heart Rate Sensor](image2)

- SpO2 sensor uses pulse oximetry for monitoring blood oxygen saturation. Hemoglobin in the Red Blood Cells (RBC) carries oxygen used for measurement. The pulse oximeter uses the two probes emitting green diode lights usually in direct contact with skin that shines the light through the skin tissue. The two led lights usually have two different wavelengths for measuring oxygenated blood and deoxygenated blood. The light absorption properties from the oxygenated blood are red at 660nm, and deoxygenated blood is near-infrared (IR) light at 940 from the light-emitting diodes [7]. The pulse oximeter determines the hemoglobin in the arterial blood that is in turn used to determine SpO2 blood in peripheral circulation [8]. Its advantage is that it is non-invasive and can easily detect changes in oxygen levels.

![Capacitive Proximity Sensor](image3)

![PPG Algorithm](image4)

![SpO2 Sensor](image5)
Fig. 4. Overview of angular velocity a 3D vector quantity with X-axis, Y-axis, and Z-axis. [9]

- The MPU6050 MEM module consists of a 3-axis accelerometer used to measure linear acceleration [1], and a 3-axis gyroscope sensor is an angular motion sensor that considers all three orthogonal axes while considering the object’s angular velocity. It also helps determine the user’s activities like running, walking, and even wrist movement. Figure 4 above depicts how angular velocity calculation equals the vector sum of \( w_x \), \( w_y \), and \( w_z \) values.

Fig. 5. An MPU6050 Pinout of MEMS MPU6050 module consisting of a 3-axis accelerometer and 3-axis gyroscope sensor

In general, the MPU6050 module measures the acceleration, velocity, orientation, displacement, and other motion-related parameters of a system object. It also contains DMP (Digital Motion Processor) used for calculation [10].

- The vibration engine in the Mi-band two contains a vibration monitor that helps monitor the wearer’s activity or inactivity and can give vibration alerts from the band and paired android device.

Fig. 6. Xiaomi-mi-max-2 vibration motor
Source: Adapted from [11]

B. Open source-based tools to be used

1. An installable catalog F-droid of FOSS (Free and Open-Source Software) applications for the Android platform shares similar functions as Google Play [12].
2. GadgetBridge is an Android application that allows usage of Huami based smart bands without the vendor’s closed source application [13].
3. Hua Fetcher: Desktop or android application that helps get tokens from Huami-based smart bands [14].
4. Node-RED: A visualization tool for Wiring the Internet of things developed by IBM Emerging Technology and Open-source community [15].

III. METHODOLOGY

A. Proposed system description

Figure 7 shows the overall view of the proposed real-time biofeedback system.

The proposed system consists of a Smart band with biofeedback sensors designed to collect related physiological data linked to the wearer’s daily activities. The band processes the sensor data and transfers it to the paired Android device. The open source-based android application Gadget bridge used to bypass the vendor’s installed software is then used to collect, display sensor results and manage the transferred data. The collected raw data is auto exported as an SQLite database for further processing and visualization based on the research needs.
The SQLite database file format used for export has many advantages [16]:

- Read and write is faster than usual file systems, thus giving it better performance.
- Its cost-effectiveness and reduced complexity with ease in the resolution of performance problems.
- Its portability across all operating systems and accessibility by other programming languages allows parallel processes to attach and read and write without affecting each other.
- Reliability due to reduced bugs due to its custom code and ability for continuous content updates and atomicity making it reliable in power failure or crash.
- It is Open source, so no limitation on use, though it is not an open contribution [16].

The conversion of SQL file format to CSV is possible based on the selected visualization technique.

For this solution, to graph, the data SQL file format with Node-RED UI is used. Node-RED is an open-source visualization tool developed by IBM Emerging Technology and the open-source community [17]. Data in the cloud sent from the android device uses Front End Node-RED (FRED) to manage multiple users. FRED simplifies and manages the instance of Node-RED, helping with complexity reduction in setting up the real-time biofeedback system [17].

![Node-RED flow](image)

**Fig. 8.** Overview of a Node-RED flow for JSON data from Android device.

To use FRED, one needs to sign up for a cloud account. Depending on data sent to FRED, one can choose how to want to process it and visualize it:

- Can create a Node-RED flow to monitor the biosensors. SQLite data is sent to FRED Node in JSON and received through the HTTP in-node that listens to the POST requests. The content type should be application/JSON for the data to be accessed. HTTP in-node parses the body of requests and makes data available to msg. payload as its properties for ease of use.
- Can use the Litedb node to process SQLite database file sent to FRED. To use Litedb, one must add a node from the storage section called the Litedb node and select how to visualize the data from the visualization category. Once the Litedb node installation completes, the next step is to click on the start instance button to start the Litedb instance. A canvas should appear that can modify to process the real-time data received from the android phone.

**B. Constraints**

- Open-source tool gadget bridge allows bonding of one device at a time for data collection and management, so for collection of data for comparison between three devices, one has to use either three different android phones for data collection or monitor one device at a time in gadget bridge meaning that each time has to retrieve the AUTH Tokens for bonding.
- Blood Oxygen monitoring in Amazfit band 5 with the SPO2 sensor using the vendor application was not working from the first day. It was tested with different individuals to see if any change could occur, but no positive changes or change in blood oxygen reading. Factors that may have affected the effectiveness of a SpO2 sensor may include poor blood perfusion [18] or wrist movement and insufficient wrist contact [19]. Traditional SpO2 sensors use fingertips as compared to the wrist. SpO2 can be affected by cold weather [20].
- Data accuracy of the Bands noticed a variation in data collected when wearing two smart bands on the same wrist regarding heart rate. PPG signals accuracy and variability in individuals can be affected by body movements related to muscles and tissue dilation, users’ anatomy, and sensor displacement [21].
- Lack of research touching on open-source tools to hack readily available biofeedback devices, especially smart bands.
- Running Node-RED locally on the android phone had its limitations Termux application cannot work with F-droid.

**C. Device hacking procedure steps**

The steps below help pair and bond the smart bands with GadgetBridge for data management and extraction [13]:

Download Vendor applications from Google Play for the Huami based bands. Zepp for Amazfit band five and Mi-fit for Mi bands five and two. Create an account and save the vendor application credentials for later use in bridging devices with Gadget Bridge. Pair up the Vendor application with the Huami smart band devices.

Download and install of F-droid android catalog on the android device, then search in the F-droid application for the GadgetBridge application and install while accepting all requested GadgetBridge permission for it to work as expected. Unrooted phone token retrieval steps for bridging GadgetBridge included selecting the Huafetcher application for token retrieval or using a python script.

Token retrieval steps using the Huafetcher application involves:

Token retrieval for Amazfit band 5 involves using the Zepp vendor application login credentials. In the Huafetcher application, the credentials are added in the email and password section and used fetch token for later use as Auth key for bonding with GadgetBridge.

Token retrieval for Mi band devices involves the selection of Xiaomi in the Huafetcher application and clicking on the
get token opens a link in the browser with Xiaomi application login displayed. The Mi-fit vendor application credentials and accepts the terms and conditions are then verified. On login opens a page with an error message. Please ignore it, copy the URL, and return it to the Huafetcher application. Paste the results and then fetch the token to bond with GadgetBridge. Copy the fetched token, proceed to the GadgetBridge Application, and click the Add (+) button to add the new devices. Ensure that Bluetooth is active for pairing. GadgetBridge should start scanning for nearby devices once when selecting start discovery.

Once the device wants to pair with is discovered, stop the discovery process. Next, proceed to long-press the device name that leads to a settings page and scroll until the AUTH Key section. Select the AUTH Key section and paste the retrieved token for the device to pair with GadgetBridge and save. The pairing process should commence next with the message displayed on the successful bonding of the band to GadgetBridge. The main page of Gadget Bridge should indicate the connection of the smart band in the App bar with various icons also visible for customization and viewing of various charts related to biosensing activities of the band visible. Customize the device and developer settings, then proceed to the data management section. The data management section offers various options for managing, storing, and exporting the SQLite database to other connected applications.

IV. RESULT

A. Data Collected

Python is used to convert the SQL data to CSV for further analysis. Figure 9 summarizes all the tables that had data and their datatypes as processed by the sensors and helps identify the user, devices used, and their attributes and metadata.

![Fig. 9. Overview of data types of all the tables with data in SQL database exported from gadget bridge.](image)

<table>
<thead>
<tr>
<th>USER</th>
<th>DEVICE_ATTRIBUTES</th>
<th>USER_ATTRIBUTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>_id</td>
<td>int64</td>
<td>_id</td>
</tr>
<tr>
<td>NAME</td>
<td>object</td>
<td>HEIGHT_CH</td>
</tr>
<tr>
<td>BIRTHDAY</td>
<td>int64</td>
<td>WEIGHT_KG</td>
</tr>
<tr>
<td>GENDER</td>
<td>int64</td>
<td>SLEEP_SEGMENT</td>
</tr>
<tr>
<td>dtype: object</td>
<td></td>
<td>STEPS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VALID_TOUtc</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEVICE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_id</td>
<td>int64</td>
<td>TIMESTAMP</td>
</tr>
<tr>
<td>NAME</td>
<td>object</td>
<td>DEVICE_ID</td>
</tr>
<tr>
<td>MANUFACTURER</td>
<td>object</td>
<td>USER_ID</td>
</tr>
<tr>
<td>IDENTIFIER</td>
<td>object</td>
<td>RAW_INTENSITY</td>
</tr>
<tr>
<td>MODEL</td>
<td>object</td>
<td>locale</td>
</tr>
<tr>
<td>ALIAS</td>
<td>float64</td>
<td>dtype: object</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MI_BAND_ACTIVITY_SAMPLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>int64</td>
<td>DEVICE_ID</td>
</tr>
<tr>
<td>DEVICE_ID</td>
<td>int64</td>
<td>USER_ID</td>
</tr>
<tr>
<td>USER_ATTRIBUTES</td>
<td></td>
<td>RAW_INTENSITY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RAW_KIND</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HEART_RATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dtype: object</td>
</tr>
</tbody>
</table>

![Fig. 10. Overview of the MI_BAND_ACTIVITY_SAMPLE table in SQLite database from the Huami based smart bands exported from GadgetBridge.](image)

- The TIMESTAMP is in milliseconds.
- HEART_RATE defines the frequency or number of times the heart beats per minute.

![Fig. 11. Sample comparison of Collected data, GadgetBridge, and Zepp application visualization of heart rate data on 29 October 2021, for Amazfit Band 5.](image)

The GadgetBridge and Zepp app data collected showed no difference in average heart rate finding the average heart-rate data from a random sample.

B. Discussion

From figure 11, the data collected from GadgetBridge did not differ much from that collected from the original vendor application indicating that sensor data was not affected by the hack. As summarized in figure nine, the data collected from the devices show the scope of how can use the biofeedback data from research. A good example is the potential to use Heart rate variability (HRV). HRV measures heart rate variations between two successive heartbeats [22]. The HEART_RATE column and TIMESTAMP can calculate heart rate variability in the band’s processed data. HRV estimates stress, oxygen uptake, cardiovascular health [17], EPOC(Excess Post Exercise Oxygen Consumption) [23], and depends on respiration, health, and genetic predisposition.
The number of sensors in the device and open-source tools shows the cost-effectiveness of using hackable biofeedback devices. When bought on their own, device sensors would be more costly in price, time, and skillsets needed to configure them to collect data. Hackable open-source devices also provide an opportunity for researchers or enthusiasts in IoT who lack skills in setting up devices to have an opportunity to use readily available open-source tools to retrieve data and build systems want to. This research shows the possibility of using hackable, readily available biofeedback devices like smart bands to collect data like vendor-locked applications. The functionality of the smart bands is constantly improving, as seen with selected devices from the Mi-band two having fewer sensors as compared to the Amazfit band 5 having the most.

V. CONCLUSION

The study shows a detailed analysis of sensors used, their limitations, how to hack the devices, and the possible use of open-source-based biofeedback solutions when developing real-time biofeedback systems. The hacked devices could collect sensor data in real-time without affecting its integrity.

The advances in MEM-related technologies in the past three years show a quick progression in technology and create an opportunity for further exploration of hackable biofeedback devices and use in real-life biofeedback simulations.

REFERENCES


