Investigating Wearable Fitness Applications: Data Privacy and Digital Forensics Analysis on Android

Shinelle Hutchinson 1, Mohammad Meraj Mirza 1, Nicholas West 2, Umit Karabiyik 1,*, Marcus K. Rogers 1, Tathagata Mukherjee 2, Sudhir Aggarwal 3, Haeyong Chung 2 and Carrie Pettus-Davis 4

Abstract: Wearable devices are becoming more and more prevalent in our daily lives as people become more curious about how well they are doing in monitoring, improving, or maintaining their health and fitness. Fitness trackers and smartwatches have become almost ubiquitous, so these devices have begun to play a critical role in forensic investigations. In this paper, the authors conducted a forensic analysis of the controlling applications for three popular fitness bands and smartwatches (i.e., Amazon Halo, Garmin Connect, and Mobvoi) on an Android smartphone device to (1) provide forensic investigators with a road-map of forensically relevant data that are stored within these applications and (2) highlight any privacy concerns that the stored data within these applications may present to the applications’ users. Our findings indicate that the three fitness applications store a wealth of user data. In particular, the Amazon Halo app stores daily, weekly, and monthly activity-related data for at least the last 13 days. The user’s Tone Analysis results were also recovered. The Garmin Connect application also records detailed user activity information, as it was possible to recover the last 15 days worth of user activity data. The Garmin Connect user’s general location was also determined via the application’s weather notification feature. Lastly, the Mobvoi application records all data points from the time the device is first used until the last time the device is used. These data points may include heart rates taken every 5 min and step counts. Our findings highlight the possibility of collecting personally identifiable information about users of these devices and apps, including their profile information, habits, location, and state of mind. These findings would be pertinent to forensic investigators in the event that these or similar applications are part of an investigation.

Keywords: data privacy; data security; fitness trackers; forensic analysis; IoT forensics; mobile forensics; user privacy; wearable devices

1. Introduction

The Internet of Things (IoT), such as smartphones, is becoming a ubiquitous technology [1]. These IoT devices have made their way into almost every corner of our lives, from transportation to healthcare, and everything in between. These gadgets have even made their way into our lives in the form of wearable devices. Wearable devices are electronic devices that can be worn as accessories, embedded in clothing, implanted in the user’s body, or even tattooed on the skin [2]. Two of the most popular types of wearable devices are smartwatches and fitness trackers [3]. Henceforth, wearables will be used to refer to smartwatches and fitness trackers in the rest of this paper, unless stated otherwise.

In recent years, fitness tracking has become popular with people looking to track their daily statistics or improve their fitness. Smartwatch sales were about 69 million in 2019 and are projected to reach more than 113 million in 2022 [4]. Their associated applications...
apps) have access to fine-grain data about persons, which need to be protected and stored securely. However, there have been instances of fitness data from wearables being leaked [5–7], exposing the user’s location, personal information, and even the location of army bases [7]. These wearables have also been used as ‘digital witnesses’ in multiple forensic investigations [8,9].

These wearable devices have been crucial in various court cases in recent years, being used to corroborate or refute witnesses’ and suspects’ claims about the events surrounding a crime. In 2018, investigators were able to use the victim’s Fitbit smartwatch data to determine her time of death, which poked holes in the suspect’s alibi [10], resulting in a conviction. Similarly, in 2021, a husband confessed to killing their wife after her smartwatch data was able to disprove their accounts of what happened [11].

There is a need to keep up with the technical changes that result from the release of these newer wearables, in order to help forensic investigators and consumers. Previous literature has focused on conducting forensic analyses of wearables with older mobile operating systems [12], computer operating systems [13,14], or the same app (Fitbit) [12–16].

The main contributions of this paper are as follows.

• The authors conducted a forensic analysis of three smartwatch/fitness tracker applications on an Android smartphone to identify privacy concerns that arise due to the way these apps store data on the Android device. The apps investigated are Amazon Halo, Garmin Connect, and Mobvoi.

• Provide the first comprehensive forensic analysis of the Amazon Halo app on Android.

• Compare the findings between Magnet AXIOM and Basis Technology’s Autopsy, taking note of any differences or advantages of using open-sourced tools over proprietary software.

• Provide a forensic roadmap to investigators tasked with examining these and similar smartwatch/fitness tracker devices.

• Identify security and privacy concerns related to the data stored by these three apps on an Android device.

The rest of this paper is organized as follows. Section 2 offers some background on previous works that have been done regarding the security and privacy of wearable devices, particularly fitness trackers and smartwatches. Section 3 details the methodology the authors propose to conduct the forensic investigation of our three chosen health apps, while Section 4 highlights our findings. Finally, Section 5 discusses the findings, while Section 6 reiterates our main points and concludes our work.

2. Review of the Literature

This section provides a review of the most relevant literature on the current study. It also discusses their shortcomings and mentions how the current study improves on those shortcomings with our work. Previous research has primarily focused on analyzing Fitbit smartwatches and the Fitbit application on various Operating Systems, including Android, iOS, and Windows. Table 1 provides a summary review of the literature and the contributions of the current study.

Williams et al. [16] conducted a forensic analysis of the Fitbit Versa smartwatch on Android 9 and iOS 12. The authors provided a clear description of the relevant artifacts recovered from these two devices after using multiple extraction methods. The authors also provided a comprehensive comparison of the forensic acquisition capabilities of Cellebrite UFED and MSAB XRY. Similarly, Yoon and Karabiyik [12] conducted a forensic analysis of the Fitbit app when using a Fitbit Versa 2 smartwatch. The authors identified the privacy concerns resulting from any forensically relevant artifacts recovered from the Fitbit app on an Android 7 smartphone device. Furthermore, the authors tested all the free features offered by the Fitbit app and the Versa 2 smartwatch, where they performed various activities with the Versa 2 smartwatch, which generated data to be recovered. For example, the authors used Alexa voice commands, tracked a workout, linked a credit card to the smartwatch, generated notifications that appeared on the Versa 2 watch screen, and set
alarms. They found that GPS locations, linked credit card information, health-related data such as heart rate and exercise activity data, and the user’s latest GPS location were recoverable from the Fitbit app. In addition, the authors noted that artifacts related to app notifications are shown on the Fitbit smartwatch, and artifacts related to any interaction with Alexa on the Versa 2 were not recovered from the Fitbit app. The methodology followed [12] is used as the basis for our current study despite the fact that the authors used an older mobile operating system (Android 7) in their work. Our research improves on this shortcoming by focusing on investigating our chosen apps on one of the newest mobile operating systems widely used today, Android 10.

MacDermott et al. [13] conducted a forensic analysis of three fitness wearable devices: a Garmin Forerunner 110, a Fitbit Charge HR, and a generic low-cost HETP fitness tracker. The authors identified forensically relevant artifacts stored by these devices by manually examining their associated Windows 10 app (Fitbit) or directly connecting the wearable to a Windows 10 computer (Garmin). The HETP band only interacts with a mobile app. However, the authors were unable to view any data that may have been stored within the app. The authors found that the Garmin device and the Fitbit Windows 10 app record a wealth of forensically relevant data. Notably, they were able to recover details about the test runs they did to populate the devices with data, including GPS locations (Fitbit), deleted records (Garmin), group chat/post interactions (Fitbit), and other user and device information (both). However, these authors did not investigate what forensically relevant data can be recovered from the mobile apps of these devices, a limitation in our opinion. As such, our investigation will extend their work by conducting a forensic analysis of the Garmin Connect app on Android 10, while Yoon and Karabiyik’s work [12] filled the gap by investigating the Fitbit app on a mobile device.

Kang et al. [15] conducted a forensic analysis of the Mi Fit and Fitbit apps on an Android 7 device after using the Mi Band 2 and Fitbit Alta HR fitness trackers, respectively. The authors showed that it was possible to recover artifacts related to the user’s entered profile information (e.g., birth date, name, weight, and height), the device used (device’s MAC address and ID), daily sleep and step records, activity data (step counts and distance timestamps), and activity tracking data (start times and GPS coordinates) from the Mi Fit app. Regarding the Fitbit app, the authors noted that they could recover the device information, step counts, sleep information, activity information, and GPS data associated with a tracked exercise. The authors noted that the sleep and activity data could be modified by the account users for both apps, while the activity data cannot be modified, but only deleted. These two observations have profound implications should these devices/apps appear during a forensic investigation. One limitation of this work was the lack of information on how the fitness trackers were populated as this process was unclear in the paper.

Almogbil et al. [14] conducted a forensic analysis of the Fitbit Windows 10 desktop app after populating the app with data from an Ionic smartwatch and Alta fitness tracker device. The authors focused on guiding forensic investigators with navigating the Fitbit data stored in the desktop app, showing how to distinguish between manually entered data and automatically logged data, and how to handle and conduct a quick but thorough analysis of Fitbit health data using open-source tools. The authors detailed their investigative methodology, which comprised of four phases: (1) Environment Setup, wherein the authors created two Fitbit accounts and signed into each on a separate virtual Windows 10 computer and the Fitbit Web app. Each wearable device was then synced with the Fitbit mobile app on a different mobile phone. (2) Feature Discovery and Data Population, wherein the authors identified all valuable features that may be useful during a forensic investigation. (3) Data Collection, where the authors obtained the Fitbit wearable data from the virtual machine files for examination and analysis. (4) Analysis, where the authors used open-source tools like Autopsy and BulkExtractor to go through the image files to identify any forensically relevant artifacts. One limitation of this work was the limited scope of the forensic investigation, as the authors only investigated the Windows 10 Fitbit desktop app.
In addition to health data obtained through a wearable device, there has been research on traditional health apps. Hassenfeldt et al. [17] conducted an in-depth forensic analysis of 13 health and fitness apps, including Strava, MapMyFitness, and Nike Training Coach, on an Android device, to identify the forensically relevant artifacts stored by these apps. The authors also discuss any privacy concerns that result from the recovered artifacts. This ethical issue has been thoroughly investigated by researchers such as Predel and Steger [18], wherein they found that data protection is one of four main issues concerning the ethical problems involved with using smart watches. The authors’ methodology involved downloading 13 apps onto an Android device, generating user data by using the apps every day, acquiring the data from the device, and analyzing the acquired data. The authors were able to recover artifacts related to the app user’s personal data, including birthdate, gender, height, weight, age, name, email, and location. Artifacts related to plaintext passwords, GPS locations, and user health data were also recovered to varying degrees from the apps the authors investigated. Two limitations or shortcomings of this work are the fact that the analysis was done using Android 6 and Android 7 devices and that the authors did not precisely detail how they populated the apps on the Android devices.

<table>
<thead>
<tr>
<th>Article Title</th>
<th>Gap</th>
<th>Objective</th>
<th>Shortcoming</th>
<th>Our Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forensic Analysis of Fitbit Versa: Android vs. iOS [16]</td>
<td>Limited wearable forensics research</td>
<td>To perform a forensic analysis of the Fitbit Versa using Cellebrite UFED and MSAB XRY</td>
<td>These authors focused on an overstudied device and app (Fitbit) and limited their tools to some of the most expensive in the market.</td>
<td>In the current study, the authors investigate a diversity of wearable fitness devices using both commercial and open-source tools.</td>
</tr>
<tr>
<td>Forensic Analysis of Fitbit Versa 2 Data on Android [12]</td>
<td>Lack of research regarding new wearables</td>
<td>To perform a forensic investigation of the newly released Fitbit Versa 2 smartwatch using Magnet AXIOM and MSAB XRY.</td>
<td>These authors also investigated the overstudied wearable controlling app (Fitbit) on an older Android OS (Android 7-Nougat) despite the newest Android OS (Android 9-Pie) being available at the time.</td>
<td>The current study improves on this research by performing the forensic investigation on a newer mobile OS (Android 10) and three different wearable fitness brands.</td>
</tr>
<tr>
<td>Forensic analysis for IoT fitness trackers and its application [15]</td>
<td>Limited wearable forensics research</td>
<td>To perform a forensic investigation of the companion apps for the Mi Band 2 and Fitbit Alta HR fitness trackers on an Android 7 device.</td>
<td>These authors again investigated the overstudied Fitbit app on an older Android OS. The authors also failed to detail the data population methodology followed.</td>
<td>The current study improves on this work by investigating a novel wearable app (Halo) on a newer and more popular Android OS. The current study also provides a detailed methodology covering all stages of the Digital Forensics Investigative process.</td>
</tr>
<tr>
<td>Digital Forensic Analysis of Fitbit Wearable Technology: An Investigator’s Guide [14]</td>
<td>Need for updated forensic analysis of the Fitbit app</td>
<td>To perform a forensic investigation of two Fitbit devices when used with the Fitbit desktop app. The authors performed their investigation using open source tools like Autopsy and Bulk Extractor Viewer.</td>
<td>The authors focused their investigation on the Windows 10 Fitbit app and did not include the mobile app nor any other popular smartwatch devices at the time.</td>
<td>The current study builds on this work by investigating one of the newest fitness bands at the time, the Amazon Halo band.</td>
</tr>
<tr>
<td>Forensic analysis of wearable devices: Fitbit, Garmin and HETP Watches [13]</td>
<td>Limited wearable forensics research</td>
<td>To perform a forensic investigation of Garmin and Fitbit fitness bands on the device level and Windows 10 app level, respectively.</td>
<td>These authors did not consider the companion mobile apps for these wearables.</td>
<td>The current study builds on this work by investigating an investigation of the Garmin Connect mobile app.</td>
</tr>
<tr>
<td>Map My Murder: A Digital Forensic Study of Mobile Health and Fitness Applications [17]</td>
<td>Lack of forensic research regarding health and fitness apps</td>
<td>To perform a forensic investigation of 13 health/fitness apps on Android 6 and Android 7 devices. The authors also investigate the privacy concerns introduced to app users due to using these apps.</td>
<td>This investigation was conducted on some of the older Android OSs at the time. Furthermore, the data population steps are not sufficiently detailed.</td>
<td>The current study explores the privacy concerns introduced to users when they use three of the more affordable, wearable fitness devices, on one of the most popular Android OSs at the time.</td>
</tr>
</tbody>
</table>

3. Methodology

The National Institute of Standards and Technology (NIST) is the premier organization for establishing acceptable measurements, standards, guidelines, and procedures for various products, systems, and technologies [19]. NIST has published guidelines on conducting mobile forensic investigations and identified the four broad steps of mobile forensics, such as:
as (1) preservation, (2) acquisition, (3) examination and analysis, and (4) reporting. The authors have followed these steps in conducting this study, as detailed below.

The authors investigate these wearable technologies from two sides: (1) identify forensically relevant artifacts from these three fitness apps to aid forensic investigators in investigations and (2) highlight any security and privacy concerns of the data being stored by these three apps on an Android device. The first side requires us to focus on what forensically relevant data can be recovered from within these apps on an Android smartphone. Mainly, the authors focus on any recovered data that may be useful to corroborate or disprove someone’s alibi during an investigation. These data may include the user’s heart rate, sleep data, activity data, GPS locations, app notifications, and voice data. The second side of our investigation requires us to identify any privacy leaks and security issues within these apps that may be troubling for these fitness apps’ users. Particularly, the focus is on highlighting any users’ personally identifiable information (PII), such as the user’s entered profile information (including name, email, and date of birth) and voice data (in regard to the Amazon Halo Band).

Before we were able to start the data population, we needed to prepare our test smartphone device. We chose to use a Samsung Galaxy A50 smartphone running Android 10 as Android 10 is one of the newer and most prevalent mobile operating systems being used today. In June 2021, Android 10 had a market share of over 36.5% among all mobile Android devices worldwide [20], making devices running Android 10 likely to be included in a forensic investigation.

The A50 smartphone used in this study was already rooted prior to beginning the investigation. The rooting process ensures that we get privileged access to the user data partition/filesystem of the device, which stores the most relevant forensic artifacts. As such, to avoid future rooting woes, we decided to begin this study without the first factory resetting the device.

To begin, we ensured that the smartphone device was rooted using the Root Checker Android app [21] after which we were ready to start device linking and data population.

Table 2 presents the smartphone device information along with the three selected wearable fitness devices. Regarding possible data types, all three wearables included multiple sensors. The Amazon Halo Band includes an accelerometer sensor, a temperature sensor, a heart rate monitor, two microphones, and a LED indicator light [22]. The Garmin vívosmart® 4 fitness tracker includes a Garmin Elevate™ wrist heart rate monitor, barometric altimeter, accelerometer, ambient light sensor, and a pulse OX blood oxygen saturation monitor [23]. The Mobvoi TicWatch S2 contains an accelerometer, gyroscope, heart rate sensor, and a low latency off-body sensor [24]. Figure 1 depicts the three fitness devices used in this study, Figure 2 highlights the steps followed in this research, starting with the design of the experiment and ending with the reporting, and Figure 3 illustrates the communication protocols and software tools used in the study. Furthermore, in the following subsections: Section 3.1 Data Population, Section 3.2 Acquisition, and Section 3.3 Analysis, we detail the methods and tools used to conduct this investigation in a forensically sound manner across the experiment phases.

Table 2. Full list of devices used in this study.

<table>
<thead>
<tr>
<th>Device Name</th>
<th>Model/Version</th>
<th>Device ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Halo Band</td>
<td>DYLAN</td>
<td>S/N: G0G13P05129200WV</td>
</tr>
<tr>
<td>Garmin vívosmart® 4</td>
<td>-</td>
<td>Unit ID: 3307917280</td>
</tr>
<tr>
<td>Mobvoi TicWatch S2 0397</td>
<td>WG12016</td>
<td>S/N: 8605X96260397</td>
</tr>
<tr>
<td>Samsung Galaxy A50</td>
<td>SM-A505G/DS</td>
<td>IMEI 1: 354463104240381</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IMEI 2: 354463104240389</td>
</tr>
</tbody>
</table>
3.1. Data Population

For this study, we actively populated the devices from 29 September 2020 to 7 July 2021. In preparation for the data population, we used a Google account that was never previously associated with any of the three apps under investigation. We then populated the smartphone device as follows:

1. Signed into the Google account on the smartphone.
2. Download and install the three necessary apps from the Google Play Store:
   • Amazon Halo: for the Amazon Halo Band
   • Connect: for the Garmin vivosmart® 4 fitness tracker
• Wear OS and Mobvoi: for the TicWatch S2 smartwatch
3. For each app, we created an account and connected the fitness tracker/smartwatch to the phone, via their corresponding apps.
4. Download and install the Whisper app [25] (to generate notifications)
5. Used the free features available to us with each wearable device to generate data. The exact type of activities we performed for each device are provided in Section 3.1.1.
6. Performed a full/physical acquisition of the Samsung Galaxy A50 smartphone with Cellebrite UFED.
7. Loaded the resultant .raw image file into Magnet AXIOM Process.
8. Examined the resultant forensic image of the Samsung Galaxy A50 smartphone with Magnet AXIOM Examine.

3.1.1. Wearable Interactions
For each app, we created an account, signed into the account, and performed the initial sync and setup of the wearable device with the smartphone. Once the wearable devices were paired with the smartphone, we performed the following activities to generate data within the apps:
• One researcher wore the three wearable devices simultaneously on their wrists each day: the TicWatch and Amazon Halo devices were worn on their right wrist while the Garmin device was worn on their left wrist. All three wearable devices were charged at least one time each day.
• We made sure to use all the features available in each app. These interactions included tracking workouts and sleep, using Alexa/Google, and generating notifications. Table 3 highlights the features tested on each wearable device, where No indicates that the feature is not available or was not tested on the respective wearable.

Table 3 suggests that simultaneously wearing multiple smartwatches might limit the results of this study as discussed in [26], however, our study does not focus on the validity of collected data from the smartwatches or does not make decisions on the quality of collected data. Therefore, simultaneous use of devices would not have any impact on the recovery of the populated data, hence does not limit the study results.

Table 3. Features populated for each wearable device.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Amazon Halo</th>
<th>Garmin vivosmart® 4</th>
<th>TicWatch S2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heart Rate</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Step Count</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Exercise Activity</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>GPS Locations</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Sleep Data</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Stress Data</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Voice Data</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Notifications</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

3.2. Acquisition
Table 4 provides a list of all the applications and software tools used during this study. Once the data population was completed, we placed the smartphone device in Airplane Mode and then forensically acquired the device using Cellebrite UFED 4PC. Cellebrite UFED 4PC can acquire a full forensic image of this rooted A50 smartphone, thereby providing us with a bit-by-bit copy of the smartphone. Placing the smartphone in Airplane Mode ensures no further data population occurs as all networks are turned off on
the device. We then loaded the image file into the Magnet AXIOM Process to allow for a more collaborative analysis process. The bit-by-bit copy is expected to contain all the user data that any installed app stores on the device.

Table 4. Applications and tools used in this study.

<table>
<thead>
<tr>
<th>Software Name</th>
<th>Version</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Halo</td>
<td>1.0.270609.0-Store_145454</td>
<td>Health Monitoring</td>
</tr>
<tr>
<td>Connect</td>
<td>4.42</td>
<td>Health Monitoring</td>
</tr>
<tr>
<td>Mobvoi</td>
<td>3.22.1-1367.602</td>
<td>Health Monitoring</td>
</tr>
<tr>
<td>Wear OS</td>
<td>2.49.0.381033832.gms</td>
<td>Linking TicWatch with Android phone</td>
</tr>
<tr>
<td>Whisper</td>
<td>9.48.1</td>
<td>Notification Generation</td>
</tr>
<tr>
<td>Autopsy</td>
<td>4.19.1</td>
<td>Examination and Analysis</td>
</tr>
<tr>
<td>DB Browser for SQLite</td>
<td>3.12.2</td>
<td>Viewing database files</td>
</tr>
<tr>
<td>Cellebrite UFED 4PC</td>
<td>7.45.1.43</td>
<td>Acquisition</td>
</tr>
<tr>
<td>Magnet AXIOM Process</td>
<td>5.2.0.25407</td>
<td>Acquisition</td>
</tr>
<tr>
<td>Magnet AXIOM Examine</td>
<td>5.3.0.25803</td>
<td>Examination and Analysis</td>
</tr>
<tr>
<td>ExifTool</td>
<td>12.30</td>
<td>Reading Exif data</td>
</tr>
<tr>
<td>DCode</td>
<td>5.5.21194.40.0</td>
<td>Forensic Timestamp Decoder</td>
</tr>
<tr>
<td>Codebeautify.org/jsonviewer</td>
<td>Web version</td>
<td>JSON viewer and formatter</td>
</tr>
</tbody>
</table>

3.3. Analysis

Once the authors had the forensic image, they examined and analyzed this image using the Magnet AXIOM Examine tool [27]. This tool allows us to view all the recovered artifacts in various formats, particularly through an “Artifacts View” or directly via the “File system View”. The examination phase simply involves us filtering any apps and date ranges that are irrelevant to our investigation, while the analysis phase involves us combing through all recovered artifacts from the three chosen apps under investigation to determine what data are recoverable, along with their locations in the Android filesystem. The authors’ goal in this study is not to determine the validity of the data being recorded by the wearable devices but rather to present an effective methodology and process for recovering and analyzing the populated data. Furthermore, for verification and validation purposes, we used Autopsy [28], which is a well-known open source digital forensics tool that is used by many agencies and practitioners around the world.

We identified potential privacy and security concerns posed during the analysis phase due to various artifacts being stored in plaintext on the smartphone device. Artifacts relating to the features investigated in Table 3 may result in the following privacy concerns to these apps’ users:

- **Direct user identification**: Artifacts related to the account and profile setup may be capable of uniquely identifying users. For instance, if possible to recover any combination of the user’s name, date-of-birth, age, home location, and email address, then these artifacts present a serious privacy concern for the app user.

- **Leak users’ possible medical conditions**: Heartrate, exercise activity, sleep data, and stress data may be able to give some indications about the user’s health and medical conditions [29,30], such as cardiovascular disease [31], diabetes [32], anxiety [33], and COVID-19 [34].

- **Facilitate user tracking**: The recovery of GPS coordinates can be used to track the user’s movements, especially should these coordinates be accompanied by timestamps. GPS coordinates also allow a malicious person to determine the user’s frequently visited locations. For example, the Fitbit app on Android 7 stores the GPS coordinates associated with an exercise the user tracked via their Versa 2 smartwatch [12].
4. Findings

All relevant artifacts being discussed in this section are recovered from the **Partition 32 (EXT-family, 112.02 GB) data** partition on the image file. This partition is represented by the first data folder in the package paths for the three apps. All artifacts discussed hereafter are recovered from the respective app’s application package as shown in Table 5. Table 6 provides a summary of all the forensically relevant artifacts that were recovered from each app. The following Sections 4.1–4.3, provide detailed findings regarding each app investigated—Amazon Halo, Garmin Connect, and Mobvoi, respectively.

<table>
<thead>
<tr>
<th>Application</th>
<th>Version</th>
<th>Package Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Halo</td>
<td>1.0.275548.0-Store_160368</td>
<td>data\data\com.amazon.healthtech.malibu</td>
</tr>
<tr>
<td>Connect</td>
<td>4.43</td>
<td>data\data\com.garmin.android.apps.connectmobile</td>
</tr>
<tr>
<td>Mobvoi</td>
<td>4.1.1-1664.650</td>
<td>data\data\com.mobvoi.companion.aw</td>
</tr>
</tbody>
</table>

Table 6. Summary of Recovered Artifacts.

<table>
<thead>
<tr>
<th>Artifact Category</th>
<th>Amazon Halo</th>
<th>Garmin Connect</th>
<th>Mobvoi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profile Information</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Heart Rate Data</td>
<td>✓</td>
<td>P</td>
<td>✓</td>
</tr>
<tr>
<td>Steps Data</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Exercise Activity Data</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GPS Data</td>
<td>N/A</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>Sleep Data</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Stress Data</td>
<td>N/A</td>
<td>✓</td>
<td>N/A</td>
</tr>
<tr>
<td>Voice Data</td>
<td>P</td>
<td>N/A</td>
<td>N/P</td>
</tr>
<tr>
<td>App Notifications</td>
<td>N/A</td>
<td>✓</td>
<td>X</td>
</tr>
</tbody>
</table>

✓: Artifact was recovered; X: Artifact was not recovered; P: Partial or limited data recovered; N/A: Not Applicable; N/P: Not Populated.

4.1. Amazon Halo Artifacts

The Amazon Halo device requires users to have an Amazon account for use. Once the device is set up, the user receives six months of Halo Membership free of charge, which includes Tone Analysis, Body Composition, Activity score and intensity, Movement Health, and Sleep scores and stages data [35]. These features are not provided to non-members. Bearing this in mind, our investigation occurred during the six-month free Halo Membership period, and thus includes the recovery of artifacts that may not be present from non-member devices.

4.1.1. User Profile Artifacts

The first thing the authors looked for were app-specific user identifiers. One of the most artifact-rich locations is the `databases\RKStorage.db` database as it holds user data, device information, and substantial activity data. This database includes one relevant table, the Table `catalystLocalStorage`. The user’s name can be found in the `currentPersonNameKey` row, while their profile information and PII, which include their gender, date of birth (birthYear, birthMonth and birthDay), height in meters, weight in kilograms and `nightRestingHeartRate`, can be recovered from the `AccessoryProfile_SyncState` row (see Figure 4). The user was asked to input all this information, expect `nightRestingHeartRate`, during the creation of the account.

The authors were able to recover a potential user ID from the `@MemoryStorage:CognitoIdentityServiceProvider.v9g8ibcn3vo2d8usc48uejoe.LastAuthUser` row within the `catalystLo-`
calStorage table. Additionally, the user’s `directed_id` (account ID) and `display_name` were recovered from the accounts table in the `map_data_storage_v2.db` database. The account ID (`directedId`) can also be recovered from the `\files\accessoriesRegistrations\registrations.json` file. The `map_data_storage_v2.db` database also contains the tables `account_data`, `device_data`, and `encryption_data`. The first two of these tables record their data as BLOBs and were unreadable through Magnet AXIOM and Autopsy. The same `catalystLocalStorage` also contains details about the Halo device being used, including the device’s `serialNumber`, `name`, and `identifier` (which is the device’s Bluetooth MAC address), and information about the firmware being used on the device (see Figure 5).

```json
{"gender": "Female", "BirthYear": 1993, "birthMonth": 2, "birthDay": 14, "deviceLocation": "Right", "height": 1.6001999487936016, "weight": 72.57472, "nightRestingHeartRate": 75}
```

Figure 4. Amazon Halo user’s profile and PII as shown in DB Browser for SQLite.

```json
{"serialNumber": "C0013D51282000W", "name": "Amazon 00W", "deviceType": "A2Y7E5283A82", "devicePresence": "ACTIVE", "batteryStatus": "DISCHARGING", "batteryScale": 100, "batteryLevel": 99}
```

Figure 5. Amazon Halo device (top) and firmware (bottom) information as shown in DB Browser for SQLite.

### 4.1.2. Activity-Related Artifacts

Table `catalystLocalStorage` also records the bulk of activity-related data. The `redux-Persist:appsync` and `reduxPersist:appsync-metadata` rows both hold varying combinations of hourly, daily, weekly, and monthly values for health metrics, such as maximum, average, and resting heart rates, sleep summaries, and skin temperatures, and activity related data, such as workouts, steps, and tone analysis results. All artifacts discussed in this section were recovered from either of these two rows, unless otherwise specified.

A session is generated whenever the user does a workout, which may be recorded manually by the user or automatically by the device and app, or uses Tone Analysis. An example of an automatically recorded workout session is shown in Figure 6, where the session ID is `35ec53fe-d316-4e05-919f-c0088538b4ba`. Along with the workout session start and end times, the authors are also able to recover session-related details such as the number of steps the user took, the number of calories the user burned in kilocalories, the length of time the user spent doing intense, moderate, and light activity in milliseconds, and the user’s maximum and average heart rates. The details associated with the session `35ec53fe-d316-4e05-919f-c0088538b4ba` are shown in Figure 7.

```
"Session": "35ec53fe-d316-4e05-919f-c0088538b4ba": {
    "id": "35ec53fe-d316-4e05-919f-c0088538b4ba",
    "version": "95cc330b-3988-48e9-8ee2-11239c5ff194",
    "type": "workout",
    "startTime": "2021-06-28T15:00:00.244Z",
    "endTime": "2021-06-28T15:12:41.893Z",
    "sessionSubtype": "Walking",
    "creationType": "Automatic",
    "sessionSummary": {
        "type": "id",
        "generated": true,
        "id": "$Session:35ec53fe-d316-4e05-919f-c0088538b4ba.sessionSummary",
        "typename": "ActivitySessionSummary"
    }
}
```

Figure 6. Session generated when the Amazon Halo device automatically detected that the user did a workout.
In addition to session-related details, the authors were also able to recover Activity and Sleep related data at varying degrees, as shown below.

- The following metrics were recorded daily for the last 13 days of our data population, from 25 June 2021 to 7 July 2021:
  - Resting Heart Rate
  - Maximum and Average Heart Rate, taken every 5 min
  - Time spent doing Vigorous, Moderate, Light, and Sedentary Activity, taken every hour
  - Calories, taken every hour
  - Steps, taken every hour
  - Sleep Summary

- The following metrics were recorded weekly for the last two weeks (28 June 2021 to 7 July 2021) of our data population, excluding Sundays:
  - Sleep Summary data
  - Activity Summary data.

- The following metrics were recorded monthly:
  - Sleep Summary data for the last five weeks (1 June 2021 to 7 July 2021) of our data population.
  - Activity Summary data for the last week (1 July 2021 to 7 July 2021) of our data population.

The row `browseWorkouts` of the sameTable `catalystLocalStorage` holds the search query and the associated results when the user searched for a workout using the criteria: **Strength**, 20–25 min, **Intermediate**, representing the category of workout, duration, and difficulty.
respectively. This query is shown in Figure 8. It should be noted that although there is a
lastSyncEpochMillis timestamp in this record, this value does not correspond with when
our tester performed the search and instead aligns with when last the app was used.

```
"queryParams": [
    {
      "type": "Workout type",
      "value": "Strength"
    },
    {
      "type": "Duration",
      "value": "20 min"
    },
    {
      "type": "Duration",
      "value": "25 min"
    },
    {
      "type": "Difficulty",
      "value": "Intermediate"
    },
    {
      "type": "Domain",
      "value": "ACTIVITY"
    }
]
```

Figure 8. Search query generated when the Amazon Halo device user searched for a workout.

4.1.3. Tone Analysis Artifacts

In order to use Tone Analysis with the Halo device, the tester was required to create
a voice profile in which she read six text samples aloud. The authors were able to recover
this recording from multiple locations within the app package, in varying formats. Two .wav
files of our complete voice profile were recovered from the \files\test.wav file and the
\files\com.amazon.sentiment\enrollment\accessory\accessory_enrolled_audio_recording_0.wav file.
The authors were also able to recover individual .wav files for each text sample our user read.
These files have a naming format of app\_enrolled\_audio\_recording\_X.wav, where X corre-
sponds to a number between 0 and 5, which is the total number of prompts shown to the user.
In addition to the video files, we are also able to recover metadata associated with this enroll-
ment period from the \files\com.amazon.sentiment\enrollment\metadata\enrollment\_metadata.json
file. This file includes the timestamps of when the Halo device was set up and when the voice
profile enrollment started and ended. This setup timestamp corresponds to the last time our
user repaired the watch connection with the phone and Halo App.

The Table catalystLocalStorage also records some data on voice-related activity. The
reduxPersist\:appsync and reduxPersist\:appsync\-metadata rows both hold data from the Tone
Analysis. An example of a session that is generated when Tone Analysis is used is provided
in Figure 9. When this analysis is done automatically, without any user intervention, the
session subtype is recorded as VbmAutoDutyCycle while the session subtype associated
with when the user does Live Tone Analysis on the app is recorded as VbmLive. The
results of the Tone Analysis from these sessions are also recoverable as we are able to get
the user’s overall positivity and energy levels during the session as well as the potential
feelings the user portrayed during the analysis. Multiple voice samples (utterances) were
analyzed and the results were recorded, such as whether the user was calm, relaxed, or
restrained, for example. These utterances are accompanied by a start timestamp in Zulu
format, the length of the sample in milliseconds, and the results (see Figure 10). Similar
data were found from the \databases\segments.db-journal file which seems to store sentiment
ratings the user may have gotten at some point from the Tone Analysis. Although this file
does not record timestamps, we were able to associate these records with records in the
catalystLocalStorage table, based on Session IDs.
Figure 9. Session generated when the Amazon Halo device user used the Live Tone Analysis feature.

Figure 10. Results for the first two voice samples of a Tone Analysis session for the Amazon Halo device user.

The utterance voice clips themselves were not recovered from the image file such that we were able to listen to them, though we suspect that they were at one point stored on the device before being deleted and subsequently overwritten. The `files\com.amazon.sentiment\accessoryAudio` folder holds a record of these deleted and/or overwritten files along with Modified, Accessed, and Created (MAC) times (see Figure 11). We saw a major difference in
how Magnet AXIOM and Autopsy handle such deleted files, as we were unable to view any recovered, deleted, or overwritten data with Magnet AXIOM Examine, though Autopsy allowed us to view the content of some of these deleted or overwritten files. For example, we were able to recover HTTP headers, URL links, and pictures associated with the Amazon Halo application and other applications on the device. During our data population, one action we took was to perform a Live Tone Analysis during a Microsoft Teams meeting to test Amazon’s claim that the Tone feature is designed to only analyze the app user’s voice based off of the voice profile created [36]. However, our observation indicate that even when our test user was silent, the Tone Analysis continued as others in the meeting spoke.

![Figure 11](image_url)

**Figure 11.** Potential deleted and overwritten utterance clips from Tone Analysis sessions along with MAC times, as shown on Magnet AXIOM Examine.

4.1.4. Cache Artifacts

The `\cache\http-cache` folder contains multiple `.0` and `.1` files of HTTP headers and videos from the Movement Assessment the user performed. This assessment was the last interaction the user had with the app and the device during our data population. The `a4f3cd88c671d58a7ad7e1b6a4e61e98.1` file within this folder stores the results of the Movement Assessment the user performed, including the session id, the session startTime and endTime timestamps, total Assessment score, various individual bodyRegionsScores, the exerciseId of the workouts the user performed along with corresponding start and end timestamps, and the exerciseId of workouts that were recommended to the user after completing the assessment. A snippet of this assessment summary for our test user is shown in Figure 12.

The `\cache\http-cache\8ddc056d858ce303e0cfdd128cc5af93.1` file was found to store a record of a video workout the user did. However, there is no mention of the video name, only a contentId (which may be the video ID), whether the user completed the activity, and the start and end timestamps. This contentId can be cross referenced with the `catalystLocalStorage` table to determine the video title and category. An example of this cross-referencing is shown in Figure 13. Furthermore, multiple folders within `\cache\image_cache\v2.ols100.1` hold `.jpg` and `.png` files of video frames and icons that are shown to the user on the app at one point or another.
Figure 12. Snippet of the results of the Movement Assessment the Amazon Halo user did.

Figure 13. Record of a workout video the Amazon Halo user did (top) and the corresponding video title (bottom).
4.1.5. App Usage Artifacts

Artifacts related to how the user used the app and device, including device and upload sync timestamps, can be recovered from the `DataPipelineDiagnosticsQueue_v1` table within the `\databases\queue-db-DataPipelineDiagnosticsQueue_v1.db` database. Similar app usage data can be recovered from multiple files within the `\files` folder. Each file is about 6.5 MB and have a naming structure as `axle-log.X`, where X represents a number starting from zero. Higher values of X represent older logs, which means that `axle-log.0` is the most recent log file. The timestamps for the last time the Halo device data was updated for various activities, such as sleep temperature, steps, and calories burned, were recovered from the `\shared_prefs\com.amazon.wellness.platform.featuregating.SharedPreferencesFeatureTreatmentStore.xml` file (see Figure 14). Similarly, the last day, week, and month that data were uploaded to the server can be recovered from the `\shared_prefs\ActiveUploadStats.xml` file, where the days and weeks are counted from 1 starting from 1 January and the months are counted from 0 starting from January.

```
<long name="ActivityRestingMr-LastUpdated" value="1625705521330"/>
<long name="AxleFeatureRestingMembershipFlag-LastUpdated" value="1625705521330"/>
<long name="ActivitySteps-LastUpdated" value="1625705521330"/>
<long name="VbmDutyCycle-LastUpdated" value="1625705521330"/>
<long name="SleepTemperature-LastUpdated" value="1625705521330"/>
<long name="VbmHistory-LastUpdated" value="1625705521330"/>
<long name="LabMemberExperience-LastUpdated" value="1625705521330"/>
<long name="ActivityCaloriesBurned-LastUpdated" value="1625705521330"/>
<string name="VbmHistory-TreatmentValue">T1</string>
<long name="ActivityLiveMr-LastUpdated" value="1625705521330"/>
```

Figure 14. Record showing the timestamps for the last time the Amazon Halo device data were updated, where the timestamp (1625705521330) corresponds to Wednesday, 7 July 2021 20:52:01 EST.

Alternatively, the `\databases\awspinpoint.db` database and corresponding journal file record details about the smartphone used with the Halo device, including the smartphone’s make, model, and carrier, and personId (the Amazon Halo user ID). Similarly, the last account that signed into the app from can be recovered from the `\shared_prefs\account_changeObserver.xml` file.

4.2. Connect (Garmin vívosmart® 4) Artifacts

The Garmin vívosmart® 4 smart activity tracker is among the many activity trackers that utilize their own app (i.e., Connect) for setup. The tracker requires users to have an account and a device to pair with. In the following subsections, we will highlight important recovered artifacts that we found in our study.

4.2.1. User Profile Artifacts

Regarding the user account information, the `\shared_prefs\mobile.auth.xml` file holds the user’s entered name, a URL link to the user’s profile picture, the user’s profile ID, and multiple credentials (tokens) that may be useful to gain access to the account (see Figure 15). Many more PII and user preferences can be recovered from the `\shared_prefs\gcm_user_preferences.xml` file. This PII consists of `UserHasCurrentPregnancy`, `primary_activity_tracker_id`, `userLocation` that is recorded as the state, `userHeight` in centimeters, `userWeightStr` in kilograms, `userName` that is recorded as the user’s email address, `userDateOfBirthString`, `userGender` and `userHandednessCapability`. Figure 16 highlights some of this user PII. In addition, the `\shared_prefs\consent_preferences.xml` file contains the user consent preferences. Moreover, the link to the profile image can be assessed. Figures 15 and 16 have been partially
redacted to preserve the privacy of our test accounts and reduce any potential compromise of these account.

```xml
object {5}
  environment: PROD
customerGUID: ca1b07d48-4fbf7ff34ee9
customerName: Elio Altorre
customerImageURL: https://s3.amazonaws.com/garmin-connect-prod/profile_images/bad7ca8f-d049-4104-89307412.png
  - connectData {3}
    userProfileID: 89307412
    displayName: 83c3f6d2c-b27d-f4ef-8f2a-bb53aba5de49
    userHasMTesterRole: False
  - credentials {2}
    - OAuthConnectData {2}
      userToken: bb621865-898b-deb447ab8848a
      userSecret: j9bbycc1-0669dx3sLOG16
    - OAuth2Data {2}
      accessToken: d76f4ad6-7e82-7156f1fead78
      accessTokenExpiresUTC: 1629497388110
      refreshToken: 043eb857-0dc9-bbab8b18c37
```

**Figure 15.** Redacted data about the currently signed in account for the Garmin Connect user, taken from the mobile.auth.xml file.

**Figure 16.** Redacted and reduced user profile information for the Garmin Connect user.
4.2.2. Activity-Related Artifacts

The `json_activities` table within the `databases/gcm_cache.db` database holds the user’s activity-related data. This table contains all the activities recorded with `timestamps`, `data_type`, and `cached_val` as important columns to investigate. When a row has `data_type` set to `ACTIVITY_DETAILS`, `cached_val` contains the JSON-formatted entry with detailed information about the activity (see Figure 17 for the formatted JSON file using the codebeautify JSON viewer). Similarly, if a row has `data_type` as `ACTIVITY_CHARTS`, the `cached_val` column will have a JSON formatted entry that contains valuable information about the activity including but limited to, all GPS location of the activity, heart rate, speed, and distance (see Figure 18 for an example). Moreover, relating back to the last weather notification, if the row has a `data_type` of `ACTIVITY_WEATHER`, in the `cached_val` column we can find details of the weather notification such as `temp`, `dewPoint`, `relativeHumidity`, `windDirection`, `windDirectionCompassPoint`, `windSpeed`, `latitude`, `longitude`, and `weatherStationDTO` which contains the ID and name of the station (see Figure 19).

![Figure 17](image17.jpg) Reduced activity details automatically recorded by the Garmin device when the user went on a walk.

![Figure 18](image18.jpg) Reduced activity chart data detailing the GPS path the user took during a walk. The GPS coordinates have been partially redacted for user privacy.

```json
"activityId": 6836406895,
"activityGUID": {
  "uuid": "ac9b4fa1-7a5-4206-9b14-84797c9945ab"
},
"activityName": "Walking",
"userProfileId": 89307412,
"isMultiSportParent": false,
"activityTypeDTO": {
  "typeId": 9,
  "typeKey": "walking",
  "parentTypeId": 17,
  "sortOrder": 27,
  "isHidden": false,
  "restricted": false
},
"eventTypeDTO": {
  "eventName": "Walking",
  "timeOnUnitDTO": {
    "startTime": "2021-05-24T15:53:22.0",
    "endTime": "2021-05-24T15:53:22.0",
    "duration": 1009.817,
    "movingDuration": 962,
    "elapsedDuration": 1009.817,
    "averageSpeed": 1.209600004907158,
    "averageMovingSpeed": 1.259740119664436,
    "maxSpeed": 1.3569999932342798,
    "calories": 90,
    "averageHR": 117,
    "maxHR": 131,
    "averageRuncadence": 99,
    "maxRuncadence": 106,
    "minActivityLapDuration": 1009.817
  }
}
```

Figure 17. Reduced activity details automatically recorded by the Garmin device when the user went on a walk.
The Garmin vívosmart® 4 is also capable of recording stress, sleep oxygen, and sleep-related data. Within the sleep_detail table inside the \databases\cache_database database, we can find the user’s sleep details. These details include date, sleep time in seconds, start sleep time and end sleep time, sleep status (e.g., light, deep, and awake), average O2 value, and lowest O2 value. Moreover, in the same database, the user_daily_summary table contains overview activity details where each row represents a day. These data cover a variety of information to the user, such as stress data, total number of steps, distance covered, calories, active time, min and max heart rate, and many more. The cache_database database contains multiple other tables, including tables acclimation_pluse_ox_details, activity_summaries, and response_cache. In addition, all the activities that are stored as JSON files in the gcm_cache.db database are instead stored as rows in the cache_database database, however, only general details about the activity as in ACTIVITY_DETAILS. The cache_database-wal file also contained relevant data that may not have yet been pushed to the .db file. Another feature of the Garmin vívosmart® 4 is its capability of showing notifications on the device. One of these notification functions is for the weather. The \shared_prefs\weatherprefs.xml file contains the weather preferences and the last time the user received a weather notification. In addition, user menstrual notifications can be recovered from the \shared_prefs\MenstrualNotificationPrefs.xml file. Other user notifications delivered through the Garmin band can be recovered from the following database: \databases\notification_database (see Figure 20 for example messages).

![Figure 19](image1.png)

**Figure 19.** Formatted, reduced, and redacted weather activity details where **name** is the name of the weather station being used.

![Figure 20](image2.png)

**Figure 20.** A reduced view of the notifications displayed on the Garmin vívosmart® 4 device.
4.2.3. App Usage Artifacts

Data about the first time the app was opened and the version code of the app were recovered from the `shared_prefs/com.google.android.gms.appid.xml` file. Additionally, the last time the band actually synced with the device was recovered from the `myfit/settings.xml` file under the `UsersLastLoginTimestamp` tag. In addition, the same file has many other information, such as how many devices are paired with the app (see Figure 21 for some information recovered from the file). Moreover, inside the `databases/gcm_cache` database, there is a table named devices, which contains information on devices that are connected to the app. In this case, it is showing the same product number and other device information found in the `androidx.work.workdb-wal` file.

Figure 21. Number of Garmin devices paired with the Connect app.

It is essential to be able to recover the band software artifacts as well as hardware information. Therefore, the Write-Ahead-Log (WAL) file at `no_backup/androidx.work.workdb-wal` has the user tracker details such as MAC Address (CE:51:6D:F9:59:A3), unit Id (3307917280), product number (2927), software version (510), and device name (vivosmart 4). Moreover, these information can be recovered from two different locations. The other location is in the phone Bluetooth configuration file at `misc/bluedroid/bt_config.conf`. Figure 22 displays the two locations. In addition, the Garmin vivosmart® 4 fitness tracker syncs, these syncs and their status and time can be recovered from Table device_sync_audit inside the `databases/sync_cache.db` database (see Figure 23 for examples). This table has sync timestamps and app version when synced; moreover, the same information can be found in `databases/sync_cache.db-journal`. Furthermore, the `files/logs/app.log` log file contains the app logs such as Sync State, Manager status, and the Handshakes with the smartphone.

Figure 22. Garmin tracker details.
Figure 23. Sync timestamps.

4.3. Mobvoi (TicWatch) Artifacts

Mobvoi TicWatch S2 requires users to first download and install the Wear OS app to facilitate the link between the smartwatch and the smartphone. Once this is done, the user can use the Mobvoi app to view the data recorded by the smartwatch. For our investigation, we focus only on the Mobvoi app to determine exactly how much data is recoverable related to the use of the smartwatch. As such, all the artifacts discussed in this section are recovered from within the Mobvoi app package.

4.3.1. User Profile Artifacts

User PII were recorded in `.xml` files within the `\shared_prefs` folder. The `account_info.xml` and `SettingSP.xml` files both contained the user’s `height` in centimeters and `weight` in kilograms. The `SettingSP.xml` file records the user’s `step_length` information while the `account_info.xml` file holds the user’s `email`, `birthday`, `age`, and a URL link to the user’s profile picture (see Figure 24 which has been redacted to preserve the privacy and security of our test accounts). The user phone number in Base64 and the IP address were recovered from the `c_64ei` and `customer_ip` rows, respectively, in the `cookies` table within the `app_webview\Default\Cookies.db` database. We were then able to recover an approximate GPS location tied to the device’s IP address from the `cache\WebView\Default\HTTP\Cache\a3780a74150574c9_0` file.

Figure 24. A redacted view of the TicWatch’s user account details.
4.3.2. Activity-Related Artifacts

In terms of health-related data, TicWatch stores these data in multiple database files. The first of these databases is the `databases\fitness.db` database. The `RECORD` table in this database stores health and activity information, such as heart rate, step count, calories, start time, end time, and distance (see Figure 25). Each entry has a `HASH` related to it, as well as a `TYPE` and a `DEVICE_ID`. It is difficult to determine the purpose of the hash value and which activities the types relate to. However, given enough time and context from other stored tables, one could potentially determine what the related health data represent. It is possible that `TYPE` identifies the exact activity that was recorded by the watch. The `DEVICE_ID` in this table corresponds to the smartwatch’s device ID. The `fitness.db-wal` file was also examined to determine whether any information was discoverable outside of the database file, but the information contained data that were already available in the associated database file.

More relevant health data was recovered from the `health_common.db` database which contains the `data_point` table (see Figure 26). The `data_point` table includes the `device_id`, which refers to either the Mobvoi Ticwatch or Samsung phone as shown by the `data_source` table (see Figure 27), a `wwid`, which may be an unique identifier for the user, a `type`, which identifies the type of activity data being recorded, and `time_to` and `time_from` timestamps, which denote the activity’s start and end times. Moreover, the `data_point` table appears to be a superset of the `RECORD` table in the `fitness.db` database, which means given these two tables, one could potentially extrapolate more meaningful information about the health activities.

![Figure 25. A reduced view of the different activities recorded in the RECORD table of the Mobvoi fitness.db database.](image)

![Figure 26. Reduced view of health-related data recorded by the TicWatch.](image)
There is a POINT table located in the fitness.db database that seems to record finer-grained and raw data collected from the smart watch before meaningful data are extrapolated from it. Columns of interest in this table include TIME, HEART, LAT, and LONG. However, the POINT table is not populated in our image. Another potentially relevant, but unpopulated, database that may hold fine-grained data is the pedometer.db database. This database contains 26 tables, including tables that can record 24-h heart rate data (rate_24_hour_table_name), blood pressure data (blood_pressure_all_data_table), and temperature data (temperature_table). These databases and tables need further investigation in future population and analysis.

4.3.3. App Usage Artifacts

The \app_webview\Default\Local Storage\leveldb\000003.log log file seems to record details on when the user visited various app activity screens and includes data such as location (city, state, and country), the name of the activity (screen) the user visited, timestamps, details about the device used, and a user ID (see Figure 28). The \shared_prefs folder contains multiple files relevant to how the app was used. For instance, the \com.google.android.gms.measurementsprefs.xml file contained information on the general use of the app, including the last time the data was uploaded to the server, the first time the app was opened, and the last time the app was paused. The wear_info.xml file contains information about the TicWatch device, including the device’s Bluetooth address (btAddress) and name (btName), Serial Number (sn), device capabilities (wearCapability), and an assigned device ID (wearDeviceID) for the smartwatch (see Figure 29). TheDeviceInfo.xml and com.mobvoi.devices_id.xml files both record device IDs, a528572b-e63e-4d02-9c6f-cb5859e9b64a and 7de2f3d1a87c56dba014029831dbf2cd, respectively, where the latter is the smartphone’s device ID.

Information about the Bluetooth devices that were paired with the smartphone was recovered from the \files\ticpod_pairing folder. This folder contains four files which record the Bluetooth MAC addresses and device names of devices that were paired with the smartphone. In our case, the first file recorded all three Bluetooth MAC addresses for the three fitness devices that we used in the study (i.e., Amazon Halo Band, Garmin vívosmart® 4, and TicWatch S2) and the next three files each held a single Bluetooth device name corresponding to a MAC address in the first file, in the same order.
Figure 29. Reduced and formatted device information for the TicWatch.

5. Discussion

This section discusses some privacy and security implications related to the artifacts we recovered from the three health apps. Most notably, our results highlight the ability for forensic investigators to recover pertinent artifacts related to the user’s location, health activity patterns, and even their state of mind at a particular time. The authors also want note that invasiveness of the forensic analysis on various data types might raise questions regarding ethics of such activities. Although it is an important and necessary concern, the focus of this research is not on the ethics of the use of smartwatches nor on the research ethics involved in this investigation. Therefore, our work is geared toward law enforcement officers that operate according to the laws of the land in which they live. Consequently, ethics is overruled by Law in the law enforcement investigation as long as legal search and seizure is performed. This is certainly the assumption in this study. As such, forensic investigations, such as the one discussed in our work, would be bound to, or dictated by, laws rather than ethics.

5.1. User Privacy Concerns

Users of Amazon Halo devices may be relieved to learn that the voice recordings used for Tone Analysis are not currently capable of being recovered from the smartphone device. However, the voice recording associated with creating the user’s voice profile (i.e., the user’s enrollment to use the Tone Analysis feature) can be recovered. This enrollment recording is a continuous voice recording, which means that there are no breaks between each read paragraph. It seems that anything that occurs in the user’s environment during this enrollment period may be captured on this recording. As such, this enrollment record may be a worthwhile last resort for forensic investigators to elicit additional evidence, assuming that the enrollment occurred within the time frame of interest.

In addition to the recovered enrollment audio videos, the Halo app also stores the user’s Tone Analysis results, which may have been initiated manually (by the user doing a Live Tone Analysis session) or automatically when the band detected the user’s voice and began the analysis. The results of this analysis may give some insight into how the person was feeling at the time of an incident, for example. However, forensic investigators would need to be careful in trusting these results, as our results indicate that during a Live Tone Analysis session, the Tone Analysis results were not solely those of the app user. Considering the Mobvoi app, the user’s privacy may be compromised due to the recovery of their entered phone number, which in itself can lead to a wealth of other user PII being discovered, including the user’s address.
5.2. Health Data Privacy Concerns

Some previous research [30] argue that profiling of users could lead to discrimination using information learned from health wearables. Due to the plethora of sensors being used and thus the multitude of varying health data points available to companies, this data may indicate more intricate details about a person’s mental or physical state. All three fitness apps investigated in the current study provide varying degrees of recoverable heart rate and activity data, potentially pointing to the wearable device user’s medical condition. Such data could and have been used to determine heart diseases and pregnancy. Consider a hypothetical situation in which wearable IoT data are used to determine a person’s acceptance for an advertised job position; a woman may be denied the job based on data from her wearable device, which may be able to predict (or infer) when a woman is pregnant, based on her resting heart rate, as research has shown that during pregnancy, resting heart rate increases by 30–50% in women [37]. Such a scenario is not far-fetched, as currently, companies refer to the applicant’s social media before deciding whether to hire the person or not [38]. Companies that have access to this level of wearable user data may also misuse the data to discriminate against their employees by denying promotions. Insurance companies may also gain access to such wearable data, particularly heart rate, activity, and sedentary data, and use this information to deny or inflate their customers’ premiums. Moreover, the availability of wearable technologies might pave the way for more frequent user/patient monitoring. However, many products can now be connected to the smartphone of the user, which can potentially allow hackers to access information by exploiting software vulnerabilities [26].

The three applications considered in this study provided varying degrees of recoverable health activity data. Regarding the Amazon Halo user’s data, a forensic investigator would be able to paint a detailed picture of the user’s daily habits for at least the last 13 days the device was used. Similarly, forensic investigators may be able to recover Garmin Connect users’ activity-related data for at least the last 15 days, in some cases. Concerning user privacy on the Mobvoi app, one of the biggest concerns is storing user PII related to the user’s health. However, it is essential to note that even though these data were found, it appears that the app developers tried to obfuscate the data so that it was not easily identifiable, though given enough time and enough data collected, it was still possible to identify some of this information.

5.3. GPS Locations

It is widespread in modern fitness trackers and smartwatches that track the GPS locations of the user to enable many insightful features that allow for a better overall calculation and improve the user’s awareness of their movements. There are two main ways these devices acquire GPS: by using the built-in GPS in the devices or using the paired phone GPS signal to determine the location when they lack the built-in GPS capabilities.

On the other hand, our study found that these apps allow us to infer the location from other pieces of data (e.g., Garmin vvosmart® 4 that gets information from the nearest weather station). Through this weather notification, the app recorded the station’s name which is near the user’s real location. Another example within the Garmin Connect app, where the activities performed by the user are named using the city/county name when the user did a walk, for example. Furthermore, the Mobvoi app recorded IP addresses that could be used to infer device locations, which can help the app to determine user location with city-level accuracy even without using or enabling GPS functionality. Although these are unorthodox methods to determine user locations, they could be essential for investigators to cross-validate the evidence found in the device. Although forensic investigators may be thrilled by the availability of such location data within these apps, Connect and Mobvoi app users may not be aware that their devices are tracking their location whenever they venture to a new/different place. This tracking may occur even without the user explicitly tracking a workout activity and thus may facilitate placing a suspect within the vicinity of an incident in question.
5.4. Filesystem Structure and Tool Performance

During the collection of data from fitness apps, it was found that many of them were storing structured JSON in single database columns rather than having separate columns to represent the data. This seems to be a typical implementation since the database libraries provided by the Android Standard Development Kit require developers to have code written to handle the migration of data every time a new column is added to the schema, which would provide significant overhead for app developers.

As a result, when using a forensic analysis tool such as Autopsy or Magnet AXIOM, they do not always successfully process data stored this way because they generally look at the column headers rather than processing the structured JSON that could potentially be stored in an individual column. Additionally, there is a pressing need for forensic tools to provide BLOB support as apps are increasingly turning to store their data as BLOBs. A significant difference between Autopsy and Magnet AXIOM is their ability to recover deleted data. As seen in Section 4.1.3 Tone Analysis Artifacts, Autopsy was able to display the overwritten voice recordings to show artifacts related to other applications on the device, while Magnet AXIOM was incapable of recovering such data. If time permits, forensic investigators may consider using multiple tools to perform their analysis, particularly if the recovery of deleted data is necessary.

5.5. Relevance to Forensic Investigations

There are several reasons why digital fitness trackers/watches are critical in digital forensic investigations. These include, but are not limited to, the prevalence of their usage, getting to know movement patterns, recovering important medical indicators, and tracking the events leading up to and after a crime. Fitness trackers for instance, have been used to assist law enforcement in various cases, including rape [39], kidnapping [40], and murder [10].

As a result, the use of digital fitness trackers/watches, when available, may help investigators and law enforcement have more confidence in (or corroborate) most of the decisions they make, which in return can support the facts presented in court during trials. In the current research, the authors recovered artifacts that investigators can use to gather more information about the user’s status, along with location data that can help the investigator understand the user’s movements and presence or absence near a specific location. The authors also recovered artifacts that can indicate the user’s health, particularly daily sedentary times, maximum and average heart rates, and sleep summaries, to name a few.

5.6. Limitations

Despite the large amount of recoverable artifacts in our work, we introduce some limitations. First, our work only concerns the three fitness tracker apps for smartwatches running on an Android 10 smartphone device. Second, due to the length of the study, some initially populated data may not have been recovered. Third, we did not populate financial information such as credit card numbers or use the device to make purchases.

6. Conclusions and Future Work

This paper offers a first-hand look at the forensically relevant artifacts that can be recovered from the controlling applications for the recently released Amazon Halo fitness band, as well as the TicWatch and Garmin fitness tracker. Therefore, these results add value to both the law enforcement and research communities. Artifacts from the Amazon Halo app were extensive, including the user’s profile information, daily, weekly, and monthly activity and health statistics, and even the results of the user’s Tone Analysis sessions. Of major significance is the inability to recover the actual voice recordings used for the tone analysis, although the voice recordings from the enrollment (voice profile creation) were recoverable. Similar user profiles and health-related data were recovered from the Connect and Mobvoi apps. Regarding GPS data, although the Connect app does store the user’s
detailed GPS locations during a walk, there is still a method by which the user’s general location can be inferred by using the weather notifications provided by the app. Similarly, the Mobvoi app user’s location may be determined from their recovered phone number within the app. Such artifacts would be crucial during a forensic investigation, where all other traditional means of ascertaining the user’s geolocation prove futile. Future work in this domain should include a thorough comparison of the artifacts in the current work with those recovered from an iOS device. Another necessary investigation would include determining whether user-deleted artifacts could be recovered from these apps.


Funding: This work was funded by the National Institute of Justice Award No. 2019-75-CX-K001 titled “AI Enabled Community Supervision for Criminal Justice Services”. Link https://nij.ojp.gov/funding/awards/2019-75-cx-k001 (accessed on 22 September 2022).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References


31. Kim, M.J. Building a cardiovascular disease prediction model for smartwatch users using machine learning: Based on the Korea National Health and Nutrition Examination Survey. Biosensors 2021, 11, 228. [CrossRef]


