Digital Forensic Analysis of Mobile Automotive Maintenance Applications

By:
Faisal Sumaila and Hayretdin Bahsi

From the proceedings of
The Digital Forensic Research Conference
DFRWS APAC 2022
Sept 28-30, 2022

Forensic Science International: Digital Investigation 43 (2022) 301440

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

journal homepage: www.elsevier.com/locate/fsidi

DFRWS 2022 APAC - Proceedings of the Second Annual DFRWS APAC

Digital forensic analysis of mobile automotive maintenance applications

Faisal Sumaila, Hayretdin Bahsi*

Center for Digital Forensics and Cyber Security, Tallinn University of Technology, Tallinn, Estonia

ARTICLE INFO

Article history:

Keywords:
Mobile forensics
Automotive maintenance applications
Forensic Triage

ABSTRACT

Mobile applications regarding automotive maintenance have gained popularity among many automobile users. These applications may store digital forensics artifacts about the mechanics and movements of the vehicles such as GPS coordinates, vehicle health reports, or speed values which can be valuable for investigating a broad category of cases such as traffic incidents, insurance disputes, or criminal offenses. This study presents the forensics artifacts that can be identified in three automobile maintenance apps and provides guidance about the required technical procedures for the extraction and analysis of these artifacts. On the other hand, although increased utilization of mobile and IoT technologies creates additional opportunities for collecting more artifacts as evidenced in this study, they will cause more burden on the backlogs of forensic labs. This study also proposes a guideline about how to align manual and logical extraction procedures with the digital forensics life-cycle that includes the triage and other follow-up steps.

1. Introduction

Mobile automotive maintenance applications provide drivers with a notification or information about the integral parts and health conditions of their vehicles so that some additional measures can be taken on time to prevent possible malfunctions. They serve as a bridge between vehicles, the internet, and mobile phones. The automotive application industry has been projected to reach a market value of $166 billion by 2025 (Alexey, 2020). As a part of this growing market, maintenance applications constitute a relatively new mobile application category that interacts with vehicles. The relevant data stored on mobile phones could prove vital for traffic, vehicle insurance, or criminal investigations, thus demonstrating a new evidence source for digital forensic investigations in such broader case categories.

These applications collect data from the Engine Control Units (ECU) located in the Controller Area Network (CAN) bus via a dongle connected to the On-Board Diagnostic-II port of the vehicle. This dongle relays the data to the corresponding applications on mobile phones via either a Bluetooth or Wi-Fi connection, thereby making them a potential source of forensic artifacts. The artifacts such as VIN (Vehicle Identification Number), speed, acceleration, GPS coordinates, and relevant temporal data could help in traffic investigations or criminal cases. The internal health information of the vehicle (e.g., braking, carbon dioxide emission) could potentially be useful in insurance investigation cases. Once the investigator links a user with a vehicle by any means, any movement of the vehicle or the vehicle usage patterns that can be deduced from the artifacts may reveal significant information about the user as well.

The digital forensics literature in the automotive domain has focused on infotainment systems, which provides more evidence about users but not the vehicles (Whelan et al., 2018). A.K. Mandal et al. researched the security vulnerabilities of Android Auto and On-Board Diagnostic (OBD-II) applications (Mandal et al., 2019). However, the forensic analysis of automotive maintenance applications has not been explored up to our knowledge.

The advances in mobile and IoT technologies have facilitated more interaction between different devices, thus, the forensic evidentiary value of such devices has increased so much as demonstrated in this study. However, this opportunity comes up with a cost in the form of an additional burden on the backlogs of forensics labs. Assigning non-specialists into field triage processes for evidence acquisitions would reduce the workload (Hitchcock et al., 2016). Therefore, it is important to identify which artifacts can be obtained by using which extraction method (e.g., manual,

* Corresponding author.
E-mail address: hayretdin.bahsi@taltech.ee (H. Bahsi).

https://doi.org/10.1016/j.fsidi.2022.301440
2666-2817/© 2022 The Author(s). Published by Elsevier Ltd on behalf of DFRWS This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

logical, or physical) so that an effective triage process could be designed.

This paper applies a systematic artifact engineering study to investigate which data of evidentiary value can be extracted from three automotive maintenance applications used in the market namely, ZUS Smart Vehicle Health Monitor Mini, Veepeak OBDCheck, and GoFar. We applied the data extraction procedures to three different car brands to see whether there exist artifact variations among them. We compared the accuracy of the data generated by GoFar with the results of the known navigation application, Waze, and the vehicle's dashboard data to shed light on the reliability of the collected data. Additionally, we proposed a data collection procedure that can be applied during the digital forensic life-cycle that includes field triage and forensic lab analysis steps.

The previous forensic research conducted in the relevant domains has concentrated on the infotainment systems of vehicles. This study addresses automotive maintenance applications which enable us to collect data about the vehicles. It is possible that the investigators may not currently know some essential details about these applications, and therefore, may not know which forensics artifacts can be collected concerning the interactions between the application user and vehicle. The mapping of extraction methods to artifacts and proposing a data collection procedure that can be applied in triage processes are the other significant contributions of this paper.

The outline of this research paper is as follows: Section 2 gives some background information about automotive maintenance applications and presents a review of the literature regarding the digital forensics studies in the respective domains. Section 3 details the methods used in the study. Section 4 presents the findings. Further discussion about the identified artifacts and the triage process proposal are given in Section 5. Section 6 concludes the paper.

2. Background and literature review

The automotive maintenance applications are installed on mobile phones and enable the user to obtain data from the vehicle by using the OBD protocol. These applications come with a dongle to connect to the car's OBD-II female port. Its primary goal is to gather and relay the vehicle's internal diagnostics. The development of the OBD standards started as a means to monitor the vehicle engine functions and gas emissions and report issues or failures in the electrical parts of the car (Zaldivar et al., 2011). The OBD-II standard was launched in 1996 and, thus, vehicles manufactured from this date have come equipped with this port (Miller, 2021). Modern vehicles have various ECUs that are responsible for sensing the environment and controlling the vehicle functions (Sim and Sitohang, 2014). The widely used communication protocol standard is the ISO 15765-4-CAN which has been mandated to be used in cars manufactured in the USA since 2008 (Electronics, 2020). This protocol is used by all vehicles for OBD-II communications with the ECU as it can reach a speed of 1 Mbps (Miller, 2021).

Automotive maintenance kits usually include a mobile application, either from the automotive application manufacturer or a compatible application available on Google Play or App Store. It links the mobile phone to the OBD-II dongle via Bluetooth or WI-FI connections (i.e., all the applications included in this study use Bluetooth connections). Once the connection is established successfully between the dongle and the phone, an existing user must log in to the application otherwise, a new account needs to be created for new users. The user configures the application by providing basic information about the car such as the make, model, year, current odometer reading, and engine capacity. The application then starts to monitor and transmit vehicle information such as acceleration, speed, GPS as well as any trouble codes that point to a faulty condition in the car. This information is presented to the driver via the application's dashboard on the phone.

In digital forensics investigations including mobile forensics cases, data extraction is performed in three ways, manual, logical, and physical. The manual extraction method involves going through the mobile phone's user interface and taking note of the information observed on the device's screen (Ayers et al., 2014). To go by the definition of NIST in 2007, “a logical acquisition implies a bit-by-bit copy of logical storage” (Jansen and Ayers, 2007). Additionally, Srivastava et al. defined logical acquisition as the method that provides the forensic investigator access to data by getting into the file system of the device (Srivastava and Tapaswi, 2015). According to the NIST Special Publication 800-101 of 2014, physical extraction can be termed as “extracting and recording a copy or image of a physical store” (Ayers et al., 2014).

There exists a considerable amount of research in the mobile forensics domain. Barmpatsalou et al. review these studies and also highlight the importance and evolution of data acquisition methods applied to mobile devices (Barmpatsalou et al., 2018a). Some mobile application categories developed for Android have been subject to detailed forensic investigations such as Instant Messenger applications, WhatsApp (Mahajan et al., 2013; Pankova, 2019), Viber (Mahajan et al., 2013) and Vault applications (Zhang et al., 2017). Some researchers also delved into possible ways of intercepting communication in mobile applications such as WhatsApp (Wijnberg and Le-Khac, 2021).

Some applications regarding the automotive domain have been addressed in digital forensic research. The forensic artifacts that can be obtained from the in-vehicle infotainment systems of Toyota and Uconnect are discussed in (Whelan et al., 2018). The application of logical and physical extraction methods showed that the infotainment system of Toyota stores more data than Uconnect. A similar study analyzed infotainment systems of four different car brands, namely Volkswagen (Passat and Tuareg), Audi (Q5 and Q7) Ford (Fiesta and Focus), and Dodge Durango (Lacroix, 2017). The security vulnerabilities of Android Auto and On-Board Diagnostic (OBD-II) have been identified in (Mandal et al., 2019), however, this study did not address the extraction of forensic artifacts.

The triage model has drawn significant attention in research communities. A model that enables the investigator to do an initial investigation on the site is introduced in (Rogers et al., 2006). As discussed by Kao et al. (2019), the increase of digital evidence in criminal cases has led to an accumulation of incomplete forensic investigations. To deal with this situation, it has become common for first-responders at an investigation scene to employ triage in an attempt to ensure only relevant devices are obtained, which in turn reduces the number of items to be examined. In contrast to the work of (Kao et al., 2019), Horsman introduced a blueprint for applying triage based on machine learning instead of human judgment (Horsman, 2021). Their main objective was the reduction of human input, hence limiting human error when giving on-site decisions. Also, Kao et al. discussed two types of digital triage; live and dead (Kao et al., 2019). In this study, live triage is defined as being conducted on devices that are still powered on by obtaining relevant information as quickly as possible and finding data that might be of evidentiary value. These prior works introduce the triage process in forensic analysis; the present paper adapts a triage process to obtain data from automotive applications on-site.

3. Methodology

We followed the guidelines of NIST regarding the order of mobile forensic extraction methods (Ayers et al., 2014). The NIST framework was adopted in this research due to its versatility. We resorted to manual, logical, and physical extraction methods for data acquisition as defined in (Ayers et al., 2014; Jansen and Ayers, 2007; Srivastava and Tapaswi, 2015).

We selected applications that use the On-Board Diagnostic II port of the car to communicate with the application, while the connection between the smartphone and the application is via a Bluetooth connection. The data extraction was done in three different brands of cars, namely Volkswagen Golf, Mini Cooper, and Toyota Corolla to identify whether there were some variations in the resulting artifact sets.

We applied manual, logical, and physical data extraction methods for non-volatile memories of Android mobile devices. We excluded volatile memory forensic methods as the focus of this study is on the artifacts related to the vehicle rather than artifacts such as passwords or other credentials.

We started the data acquisition step with the manual extraction method, then progressed to the logical and finally applied the physical one. This order applies the logic of moving from non-intrusive to intrusive methods due to the possibility of losing data or damaging the phone if the investigator does not have the required expertise or apply the right techniques when performing the more intrusive extraction methods. This order supports the idea of giving the initial investigation tasks at the triage process to the non-specialists and moving forward to other steps when needed.

3.1. Tools and devices

We conducted the forensic investigations on the machines with the OSs, Windows 10 Home 64-bit (10.0, Build, 19041), and Kali Linux (version 2020.4) on VirtualBox 6.1. All the tools that were used for forensic extraction and analysis are listed in Table 1. We utilized the open-source Autopsy (4.16.0), as the main forensic analysis tool.

3.2. Data collection

The details of the mobile automotive maintenance applications are given in Table 2.

We used the mobile phones, Xiaomi Redmi 9 (version MIUI 11.0.8.0 QJCEUXM), Samsung Galaxy A20e (SM-A202F), and iPhone XR (iOS 14). The latter two were used at the stage of validating the accuracy of data obtained by GoFar. The outline of the data collection procedure is given in Fig. 1.

Each application has a dongle that connects to the OBD-II port of vehicles. ZUS Smart Vehicle Monitor mini and GoFar have their mobile apps and dongles. Veepeak OBDCheck comes with a dongle and mobile app that does not provide a dashboard. However, it is compatible with the OBD2 scanner application, Car Scanner. Therefore, we used Car Scanner as a dashboard alternative for this case.

Bluetooth is the wireless communication protocol between the dongle and the smartphone for each application. The applications save data on the cloud so that users can access the same dashboard information when they use another phone. However, users can not use any service directly given in the cloud as any user access interface is not provided by the vendors. Thus, the artifacts that can be obtained via cloud forensics are considered out of scope in this study.

Table 2
Mobile automobile health maintenance applications.

Applications	Model/Build	Version
ZUS Smart Vehicle Health Monitor mini	ZUHMBKBTV	7.1.0_70102
Veepeak OBDCheck Bluetooth OBD2 Scanner	OBDCheck BLE+	1.75.9
GoFar	Model 3	2.4.17
Car Scanner	400765	1.76.5

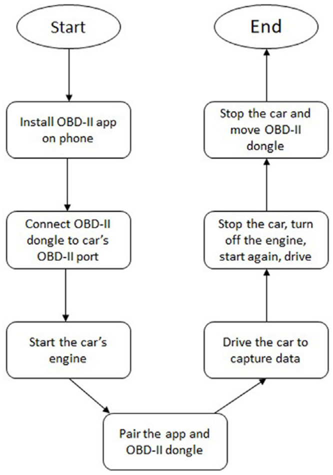
Start
Install OBD-II app on phone
Connect OBD-II dongle to car's OBD-II port
Start the car's engine
Pair the app and OBD-II dongle
Drive the car to capture data
Stop the car, turn off the engine, start again, drive
Stop the car and move OBD-II dongle
End

Fig. 1. Data collection process.

We created artifacts using Volkswagen Golf (2012), Mini Cooper (2018), and Toyota Corolla (2020) vehicles. We drove these cars between the areas of Mustamäe and Kristiine within the city of Tallinn usually after midnight to avoid traffic situations and have the flexibility to move at different speeds. The tests were done within four (4) timelines; February 25th, March 9th, March 12th, and March 16th in 2021.

We installed the mobile applications of the automotive kits on the Xiaomi Redmi Note 9 smartphone and connected the particular dongle to the car's OBD-II port. We started the engine of the vehicle and then paired the automotive application on the smartphone to the dongle via Bluetooth. After establishing a connection, the vehicle was driven around for an average of 3.8 km. Lastly, the car was parked and the engine was turned off once the trip was completed. The OBD-II dongle was then removed.

These steps were repeated for all 3 applications. The GoFar and

Table 1
Forensic tool set.

Tool	Version	Purpose
Android Debug Bridge (ADB)	1.0.41	Communicating and interacting with the Android Phone
Android Backup Extractor	V20210224105130	Packing the extracted backup file
Apache Maven	3.6.3	Packing and unpacking jar files
SQLite Browser	3.12.1	Reading database files
Autopsy	4.16.0	Analyzing the forensic data
Xiaomi Mi_Unlock Tool	4.5.813.51	Unlocking the bootloader of the Xiaomi Redmi 9
Magisk Manager	V20.4	Rooting the smartphone
BusyBox	1.32.1	Setting up a TCP connection between computer and smartphone

Veepeak OBDCheck were tested on all 3 cars, however, the ZUS smart vehicle monitor mini was able to connect to only the Volkswagen Golf. The manufacturer of the device advised that the device works better with a single vehicle and will therefore have connectivity issues when trying it with multiple cars. This prevented us from testing the device on the Mini Cooper and Toyota Corolla.

4. Results

In this section, we report the main findings of each extraction effort, namely manual, logical and physical. After presenting a general review of the artifacts obtained from the applications, we give detailed guidelines about the technical procedures applied for each extraction method in separate subsections.

The logical extraction is done in two modes, unrooted and rooted. Although we did not identify additional artifacts by applying rooted logical and physical extraction, we completed these tasks for the sake of completeness of the analysis.

We present the results of each extraction for all applications in Table 3. Each row of this table represents a distinct digital forensic artifact and the columns indicate whether the corresponding artifact was obtained from the analyzed applications, if so, what the extraction method was. A summary of the number of artifacts that were obtained per extraction method is depicted in Fig. 2.

As given in Fig. 2, the number of artifacts that can be expected from the applications, considering both extraction methods, manual and logical, may vary.

We did not observe variations between the artifact sets obtained from different car brands except that, in the case of the Toyota Corolla (Hybrid) car, the Veepeak dashboard provided additional artifacts such as Hybrid Battery Current and Hybrid/EV Battery System Voltage. The artifacts collected from each car brand by using any extraction method are given in Table 4. In total, GoFar provided more artifacts when compared to other applications. The high number of artifacts obtained in the logical extraction of GoFar is noteworthy. We identified encrypted files in Veepeak, which may be the reason for the low number of artifacts in the logical extraction of this application.

Some key findings can be derived from Table 3 when the artifact types of each application are reviewed in detail. Only Veepeak does not provide GPS coordinates of the trips. However, this app enables the specialists to manually extract various artifacts about mechanical parts of the vehicle whereas ZUS presents relatively fewer artifacts about these parts (e.g., braking, fuel consumption, acceleration). GoFar provides more complete data, including more details about the user, when both extraction methods are considered.

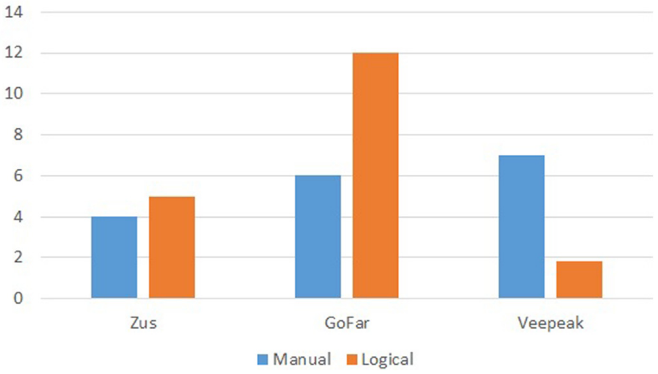

Fig. 2. The number of extracted artifacts per acquisition method.

It can be observed that the manual extraction yielded valuable data via their respective dashboards, the artifact set still varies though. A specialist in the field can get the maximum speed and RPM values from the dashboards of ZUS and Veepeak, which would be important for traffic investigations. The traveled distance can be easily accessed by manual extraction in Veepeak and GoFar. GPS locations can be gathered in GoFar and ZUS. On the other side, critical attribution data for the vehicle, VIN, can be only accessed by logical extraction in ZUS and GoFar, Veepeak does not provide that data at all. The date and time of the trip can be obtained by logical methods in all applications, only GoFar presents that data via its dashboard.

4.1. Manual extraction results

The dashboards of the applications can be considered the main interface for implementing the triage process in the field. It is mostly expected that non-technical specialists do the initial checks on the dashboards and try to understand whether an initial finding is enough for the clarification of the case or should be escalated to a further step.

Observing any data on the ZUS application dashboard was not possible when the OBD-II dongle lost its connection with the smartphone. However, it should be noted here that this data is still stored in the log files which can be obtained using the logical

Table 3
The summary of artifact extraction.

Artifacts	ZUS		Veepeak		GoFar	
	Manual	Logical - Unrooted	Manual	Logical - Unrooted	Manual	Logical- Unrooted
VIN		X				X
Vehicle Name/ID	X	X	X	X	X	X
User ID						X
User email						X
User Name and Surname						X
Date and Time (Trip)		X		X	X	X
GPS Coordinates of the Trip	X	X			X	X
Distance Traveled			X		X	
Maximum Speed	X	X	X			X
Average Speed			X		X	X
Maximum RPM	X	X	X			
Fuel Consumption			X		X	
Acceleration			X			X
Braking						X
Carbon Dioxide Emission					X	
Vehicle Make & Model						X

Table 4
The Summary of Artifacts obtained from Different Car Brands.

Artifacts	ZUS	Veepeak			GoFar		
	Volkswagen Golf	Volkswagen Golf	Mini Cooper	Toyota Corolla	Volkswagen Golf	Mini Cooper	Toyota Corolla
VIN	X				X	X	X
Vehicle Name/ID	X	X	X	X	X	X	X
User ID							
User email					X	X	X
User Name and Surname					X	X	X
Date and Time (Trip)	X	X	X	X	X	X	X
GPS Coordinates of the Trip	X				X	X	X
Distance Traveled		X	X	X	X	X	X
Maximum Speed	X	X	X	X			
Average Speed		X	X	X	X	X	X
Maximum RPM	X	X	X	X			
Fuel Consumption		X	X	X	X	X	X
Acceleration		X	X	X	X	X	X
Braking					X	X	X
Carbon Dioxide Emission					X	X	X
Vehicle Make & Model					X	X	X

extraction process. Therefore, an active connection between the automotive application and the smartphone is required when the data is accessed via the dashboard. This is an important observation in terms of the forensic triage procedure as it would mean that, depending on the mobile application and the dongle, an investigator needs to ensure that the OBD-II dongle is not pulled out and there is close proximity between the OBD-II dongle and the smartphone. We were able to obtain vehicle speed, GPS coordinates from the start point to the endpoint of a trip, engine coolant temperature, and tire pressure from the ZUS dashboard. This application was tested only on the Volkswagen Golf due to its limitation as stated before.

The content of data on the dashboard of the GoFar application was the same for all three cars. In contrast to the ZUS application, an active connection was not needed to browse through the dashboard. The main page of the dashboard lists all vehicles and their data. In addition, each trip's log file in.csv format can be exported by clicking on the “share” (i.e., sending via email, Bluetooth, or saving on Google Drive). The artifacts obtained from each car for GoFar are as follows: (a) total distance traveled in kilometers, (b) an estimate of the average gasoline consumption measured in kilometers per liter, (c) the average vehicle speed, (d) the total time traveled for the trip, (e) amount of carbon dioxide emissions in kilograms during the trip, (f) the estimated cost of gasoline used during the trip, (g) a map of the distance traveled including departed and arrived destination.

Similar to the GoFar application, we were able to extract information from Veepeak OBDCheck without the need for an active connection between the dongle and the mobile phone. The data such as distance traveled, speed, fuel consumption, acceleration, etc. regarding each trip per vehicle is stored. The recorded information is also stored by the date and time of the trip. In addition to individual travels, the following statistics about each vehicle can be obtained: average fuel consumption, average speed, distance traveled, and used fuel amount.

4.2. Logical extraction results

In the logical extraction part, we traced the files in the application folders created on the android phone. In the first attempt, we used an unrooted phone. We also applied a logical extraction technique to the rooted device.

ADB Backup is considered the main approach at this stage (Lukito et al., 2016). Firstly, we activated USB debugging on the phone to access the developer options and then installed Apache-maven version 3.6.3 as part of the Android Backup Extractor obtained from the GitHub repository (Elenkov). Next, we connected the smartphone to the machine via a USB cable and executed the ADB backup command and the resulting file was packed into.tar format and then extracted. The files and the forensic artifacts identified at this logical extraction stage of all applications are given in Table 5. The findings are summarized as follows:

Item 1: The log file extracted from the ZUS application folder contained location data as well as the connection details of the OBD-II dongle's pairing with the android smartphone. The GPS data showing the start and final points of the vehicle movements could be useful in an investigation.

Item 2: The ZUS application saves the vehicle VIN in this file. The VIN, for the sake of clarity, is the vehicle identification number that is used to specifically identify the car. A lookup of the VIN on a VIN checker online (e.g., https://uk.vin-info.com/) shows some basic information about the vehicle to confirm the validity of the VIN. This data could be useful during an investigation as it uniquely identifies the vehicle.

Item 3: This file found in ZUS contains the maximum speed that the vehicle had during the tests. The maximum speed information displayed on the dashboard (39 mph), which is obtained during the manual extraction phase, was less than the value of 51 mph given in the file. It is crucial to note that we observed a lag of about 1–2 s between the temporal value of the actual speed and the information displayed on the dashboard (i.e., explained in detail in Section 4.4). This slight delay in information being relayed to the dashboard, as explained by the device manufacturers, is dependent on the computing power of the vehicle's ECU (Nonda, 2321). The maximum speed information could be useful in traffic dispute cases.

Item 4: Veepeak saved its connection details in this log file in text format. It contains the start date and time of the trip. Additionally, detailed information about the functionalities such as adaptive headlights, navigation system, rain light sensor, parking brake, keyless ignition, and emergency response unit of the vehicle was captured in this file. This information could be useful in an insurance investigation where the functions of the car are being sought to determine the state of the car before an accident.

Item 5: The GoFar application stores the date and time of connection initiated by the OBD-II dongle for a trip. It also shows the make and model of the mobile phone that the application is installed on. The information regarding the smartphone that the

Table 5
Identified artifacts from logical extraction.

Item	App	Folder/file Path	Acquired Data
1	ZUS	\apps\us.nonda.zus\ ef\ZUS\Log	GPS position, the details of the OBD-II dongle, and vehicle
2	ZUS	\apps\us.nonda.zus\sp \FILE_NAME_VEHICLE	VIN Number
3	ZUS	\apps\us.nonda.zus\ sp\ezzy_saver_sp.xml	Maximum speed, maximum RPM, and Boolean value showing that the OBD recorded the values
4	Veepeak	apps\com.ovz.carscanner\f	Start date and time dongle connected to the car, Vehicle name
5	GoFar	\apps\co.gofar.gofar\f	Date and time of connection, mobile phone model, VehicleID, UserID, Android version on the phone
6	GoFar	\apps\co.gofar.gofar \f\default	VIN, VehicleID, userID, email, first name & last name, GPS, speed, average speed, acceleration, braking, vehicle make & model

log showed was “androidModel”: “Xiaomi M2004J19C”. This log file also contains VehicleID and UserID information, which are unique identifiers tagged by the application for each vehicle the application was connected with.

Item 6: The REALM file found in the GoFar application folder has rich content. Using a REALM browser, we were able to read the file content which covered the VINs of two vehicles (Mini Cooper and Toyota Corolla). The REALM file stored the user information (email, first name, and last name) that was provided during the initial setup of the application. We also found the country and the locations that the car traveled to. The average speed and instant speed for every 2 s were recorded in this file. The user data such as email address and name can help identify the user and provide some level of attribution.

After the logical extraction from the unrooted device, we followed the steps outlined in (Singh, 2020) to perform the rooting of the Xiaomi mobile phone. We had to unlock the bootloader, restore the backed-up data, root the phone, and finally proceed to extract the artifacts from the phone with the root privileges. We downloaded the Redmi 9 firmware MIUI 11.0.8.0 from the official Xiaomi site, obtained the ADB & Fastboot driver and the Magisk Manager APK as mentioned in (Singh, 2020), and proceeded to download TWRP. Next, we downloaded the vbmeta.img file and then booted into recovery mode to install the Magisk application to root the phone. After the phone had been rooted, we proceeded to obtain the*/Android/data* folder which provided us with the subfolders including the automotive applications folders by utilizing the ADB PULL command. We did not identify an additional important artifact from the maintenance applications when compared to the results of logical extraction done for the unrooted phone.

4.3. Physical extraction

We extracted the image of the file system from the rooted Xiaomi Redmi 9 smartphone. This device provided the required privileges to install the BusyBox application via the Magisk Manager application. After connecting the phone to the computer for the transfer of the phone's image via a TCP connection, we ran the ADB shell command from the Windows command prompt to gain shell access to the phone and then listed the partitions on the mobile phone with the command cat/proc/partitions. We copied the physical disk of the phone. After analyzing the image files, we were not able to find additional artifacts.

4.4. Validation results

In Sections 4.1 and 4.2, we identified that a significant set of artifacts such as GPS locations, speed, and relevant timestamps can be obtained from automobile maintenance applications. This section presents the results of the validation tests regarding the forensic artifacts obtained from the GoFar application. More specifically, we investigated the accuracy of artifacts by comparing them with the data generated by a known navigation application, Waze, and the vehicle's dashboard data. We selected GoFar, as it provides a rich set of forensic data, and checked GPS and the route information obtained from the dashboard and speed information from the default.Realm file.

We used 2 additional smartphones, a Samsung Galaxy A20e and iPhone XR, for these tests. We installed the free, yet popular navigation application, Waze on the A20e and then recorded the screen which displayed the route of the test drive from the starting point (E. Vilde tee) to the destination at Rimi Supermarket (Sopruse pst). The iPhone XR was used to record the vehicle's (Toyota Corolla Hybrid) dashboard to capture the speedometer readings. We then compared the values (speed, GPS, and timestamps) obtained from the GoFar default. Realm file with the outputs recorded on Waze.

We observed that Waze showed the starting point as E.Vilde tee which was similar to the one that the GoFar application dashboard recorded. The map data of GoFar shows the starting address is E. Vilde tee 117 and triangulates this location to a 500-m radius. The Gofar application started recording the trip once the dongle established a Bluetooth connection with the car at 19:05 (07:05 PM). We started moving from the initial point at 19:08 (07:08 PM). We also noted that the destination point, Sopruse pst, was captured by both Waze and GoFar (showed Sopruse pst 174). As in the case of the starting point, GoFar states the address with the zip codes. The arrival time in the background of the A20e phone that ran the Waze application showed 19:17 (07:17 PM). The GoFar dashboard showed 07:20 PM, which was the time we switched off the car's engine and ended the connection between the dongle and the car. The Gofar application logged the end time of the trip when the dongle was disconnected from the car.

Furthermore, we checked the speed data captured by the Waze and GoFar application and correlated it to the Corolla's speedometer reading at a particular time (7:12 PM) as an example. We recorded the speedometer of the Corolla on this trip.[1] Both the Waze application and the vehicle's speedometer recorded 52 kM/h at this specific time whereas the GoFar application showed 51 kM/h. We were able to get the same speed data from Waze, the speedometer of the car, and the GoFar logs for the times, 19:09, 19:13, and 19:14. The speeds obtained across all three interfaces were 50 kM/h, 38 kM/h, and 50 kM/h respectively. However, the speed data we observed at 19:15 had a variation of 1 kM/h between the GoFar log entry and that of the vehicle's speedometer and Waze. Both Waze and the vehicle recorded speeds of 50 kM/h at this time, but the GoFar default.Realm file showed a speed of 49 kM/h.

The results indicate that the application, GoFar, records and shows the vehicle data although not exactly in real-time. The ZUS manufacturer explains that the vehicles use a different variation of OBD2 protocols and ECUs have varying computer powers, thus,

[1] https://youtu.be/hKhciSXPZlc.

these factors may induce data returning rates delays up to 1 or 2 s, which can be reflected on dashboard data (Nonda, 2321). Such time variations may result in changes in the mapping of time to speed.

5. Discussion

Up to our knowledge, the forensic analysis of automobile maintenance apps has not been reported in the literature. The existing research in this domain has focused on gathering evidence from infotainment systems (Whelan et al., 2018). Although these systems may give significant forensic evidence about the users, the present study targets the vehicle as the main subject so that the benefit of the digital forensic analysis can be extended to other investigation types such as traffic incidents or vehicle insurance cases.

As can be expected in most mobile applications with relatively simple data collection and analysis functions, the evidence sources in automobile maintenance applications can be found in some application files that can be accessed by using simple manual and logical extraction methods. The logical and physical extraction methods requiring rooting the device did not provide additional artifacts in this study. However, physical extraction is recommended when data recovery is needed. Nevertheless, the cases involving these applications can be solved with basic or moderate-level technical capability. This may lead us to two significant implications.

First, if the non-technical specialists taking a role in the triage process can be equipped with forensic tools that can fulfill the logical extraction and data analysis functions at least semi-automatically, then such specialists can complete the logical extraction step in the field when necessary after manual extraction. Thus, the number of cases that may be escalated to the forensic labs could decrease enormously. Here, the burden regarding the tool vendors is how to do artifact engineering of different applications and incorporate the findings into the tool to cover wide applicability.

Second, it is possible to benefit from the forensic analysis of automotive maintenance applications in a high number of cases even in the sectors, for instance, insurance which may not have advanced technical capabilities but could provide services to some extent. On the other side, it is important to note that insurance companies tend to increase their forensic capabilities due to some incident handling services they launch within their cyber insurance products. They can utilize those capabilities in other insurance products requiring the investigation of mobile and IoT devices.

Although we found that relatively low technical capability can be enough for the investigation of maintenance applications, encryption may still constitute a bottleneck, which could make the investigations more complicated in some cases. We identified that Veepeak encrypts some files (with.brc extensions), which prevented us from doing further analysis. This is the reason why we could not extract much data from this application by using the logical method. Although the decryption of these files could be possible by using some advanced techniques (i.e., identifying the encryption credentials via reverse engineering), we did not cover this work within the scope of this presented study.

Furthermore, it is important to underline that mobile devices upload data to the cloud for various reasons, indicating that cloud forensics would be a significant evidence source in such applications (Barmpatsalou et al., 2018b). The applications covered in this study save their data to the cloud but did not provide us an interface to access this data. Obtaining artifacts from the cloud may be relevant in many cases where an investigator is not able to obtain information from the mobile phone due to it being damaged, or if the application has been deleted from the phone and the physical extraction is not able to retrieve the deleted application. An investigator may reach out to the device manufacturers with the right legal warrants and seek any assistance in accessing the information on the cloud.

We found 1–2 s lag between the information captured by the automotive application dashboard of GoFar and the actual speed of the vehicles. It should therefore be noted that the top speed showing on the application's dashboard may be slightly higher in the actual sense. Although we do not provide a complete accuracy analysis for all applications in this study, we contemplate that the research communities should focus more on the assessment of data generation accuracy in relevant forensic domains (e.g., mobile and IoT forensics) besides the forensic extraction efforts. This is especially important in applications that capture physical actions.

Finally, we present a guideline, as outlined in Fig. 3, about how manual and logical extraction methods can be mapped to the different stages of the digital forensic handling process that also includes a triage stage. For clarity of our guidelines, we have devised a scenario in which an investigator has been called to the scene of a traffic incident. We assume that a vehicle using an automotive application moving on a secluded road at a speed unknown to the investigator crashes into another car upon a turn. Upon arriving at the scene, the investigator should check if the OBD-II dongle is still connected to the car's port. If there is a connection, a manual extraction can be done on the mobile phone (Step 1), but in the scenario that the connection has been severed, the investigator can check if the application dashboard is still accessible (Step 2). If the dashboard can be accessed, then manual extraction can still be done. However, if the application provides no information due to the dongle being disconnected, the smartphone should be bagged and sent to the lab for further analysis (step 3). After manual extraction has been done and it is ascertained that enough evidence has been collected, the investigator documents the evidence and collects proofs (step 4) otherwise, the smartphone is then sent to the lab for detailed analysis (step 3). At the forensic lab, a logical extraction is then performed on the phone (step 5). In a situation where the investigator believes more evidence is needed, then a physical extraction is performed (step 6) to check for possible deleted information about the automotive application. If the mobile phone has been damaged, the investigator can use other methods such as reaching out to the device manufacturers for accessing the cloud data.

The provided guideline helps the forensic analyst make decisions at the scene of the incident. This triaging can help minimize the need to send seized devices directly to the forensic laboratory for analysis and can prove useful in cases where time is of utmost importance. As the use of automotive applications is becoming prevalent, this guideline complements the study which provides a comprehensive view including artifact engineering and data accuracy assessment.

Several studies have developed frameworks to incorporate triage into forensics investigations. A plethora of models or frameworks have been developed for evidence collection in digital forensics and most of them have incorporated triage (Thakar et al., 2021). The authors introduced a framework that inculcated a triage at the scene of the crime and how the evidence collection or extraction should be done. The proposed guideline in the present study adopts a similar idea to the automotive maintenance applications.

Data integrity and legal considerations should be taken into account concerning the extraction methods. It is a known fact that forensic investigators collect artifacts that can be presented as valid evidence in the prosecution of a court case. The integrity of such data is of crucial importance to a case as any tampering may render the evidence inadmissible. Mislan et al. stated that browsing

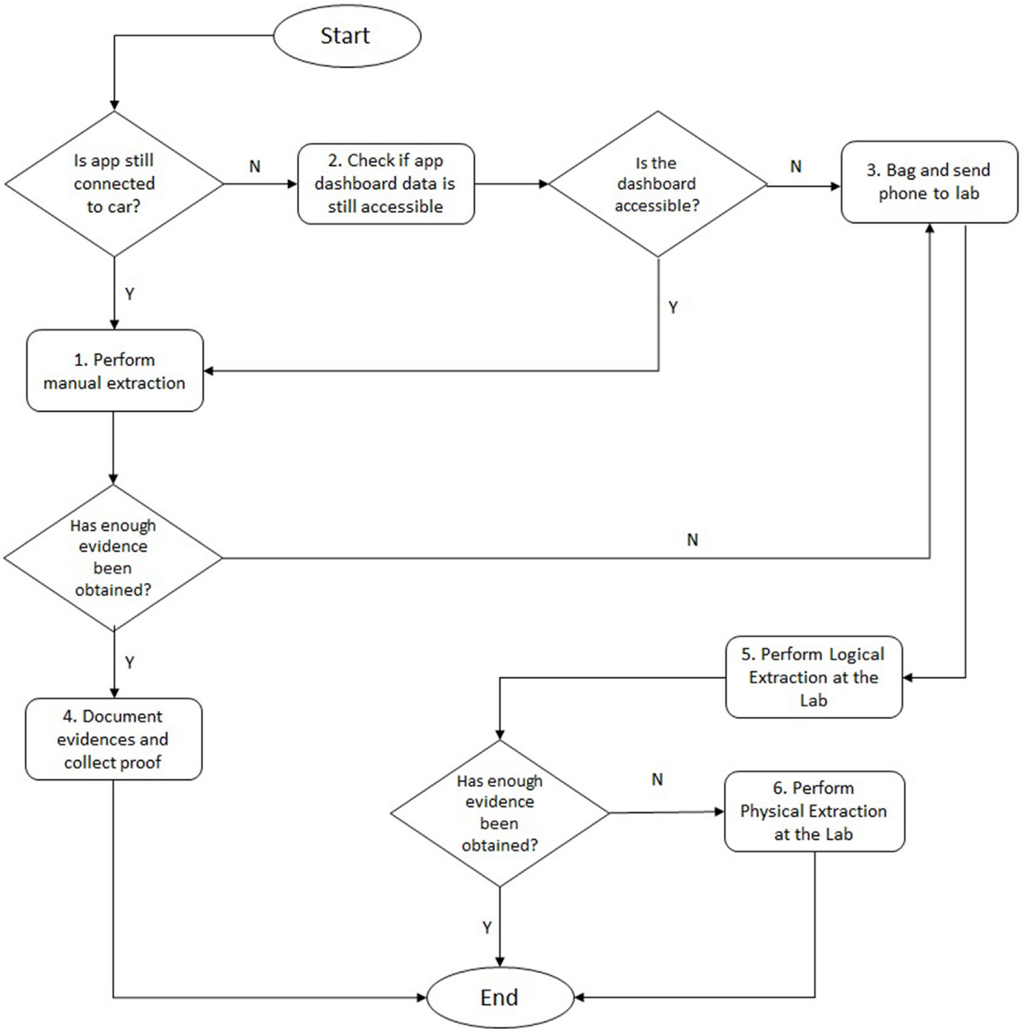
Start
Is app still connected to car?
N
2. Check if app dashboard data is still accessible
Is the dashboard accessible?
N
3. Bag and send phone to lab
Y
Y
1. Perform manual extraction
Has enough evidence been obtained?
N
Y
5. Perform Logical Extraction at the Lab
4. Document evidences and collect proof
Has enough evidence been obtained?
N
6. Perform Physical Extraction at the Lab
Y
End

Fig. 3. Sample procedure for data collection for a traffic incident.

through the mobile phone manually for artifacts is not recommended if the investigator is not forensically adept at examining the phone (Mislan et al., 2010). Thus, investigators should use tools that enable them to collect artifacts without necessarily browsing through the phone during the manual extraction. Additionally, documenting all the steps while conducting the manual extract is advised (Mislan et al., 2010).

We presented a guideline that shows the steps to examine and collect artifacts from the mobile phone based on one scenario, which is a traffic incident. However, there may be legal issues that should be considered when conducting, for instance, the manual extraction upon arriving at the scene. There is the question of what the applicable law is in regards to the seizure or collection of the mobile phone by the investigator for immediate examination. We did not cover such legal questions in this study. Thus, the proposed guideline assumes that there is a valid legal right for the investigator to collect and analyze the mobile phone.

Anti-forensics techniques that can be applied by the app user would also constitute an obstacle in our case. Users can simply resort to uninstalling the app and wiping the relevant remnants beside other advanced techniques (Garfinkel, 2007). We acknowledge that it is almost impossible to identify these cases at the triage step in which manual extraction is utilized by non-technical investigators without any tool support. Depending on the criticality level of the case and the technical capability of the user, such investigators may escalate the analysis to the forensic labs. If they are equipped with a triage forensic tool at the scene, it may be possible to do some pre-assessment for the detection of anti-forensics actions (e.g., similar work is done for windows machines in (Park et al., 2017)). However, full assessment can be achieved at forensic labs.

6. Conclusions

Automotive maintenance applications are emerging software that comes with OBD-II dongles to help monitor the health of a vehicle as well as track the traveled routes and fuel consumption. As the market for these applications is projected to rise, we explored the forensic artifacts that could be extracted from three maintenance applications, ZUS, GoFar, and Veepeak. We used mobile forensic techniques; manual, logical, and physical extraction methods to extract and subsequently analyze the data. Additionally, we compared the accuracy of GPS and speed information obtained from GoFar with the results of Waze and a vehicle's dashboard information.

We observed that the ZUS and GoFar save the VIN of the vehicle and provide GPS data whereas we did not identify those artifacts in Veepeak. Overall, the GoFar application provided us with the most information in terms of numbers at both manual and logical acquisition phases. In this study, we also draw attention to the validation of data transmitted between the cyber-physical components and mobile apps. We identified that speed information captured by GoFar may have slight variations when compared to other sources. Finally, we proposed a data collection procedure that can be utilized during the triage and follow-up stages of a forensic investigation.

References

Alexey, Chalimov, 2020. Connected carts: TOP 5 IoT automotive apps and how to develop one [Online].Available. https://easternpeak.com/blog/connected-cars-top-5-iot-automotive-apps-and-how-to-develop-one.

Ayers, R., Jansen, W., Brothers, S., 2014. Guidelines on mobile device forensics (NIST special publication 800-101 revision 1). NIST Spec. Publ. 1 (1), 85.

Barmpatsalou, K., Cruz, T., Monteiro, E., Simoes, P., 2018a. Current and future trends in mobile device forensics: a survey. ACM Comput. Surv. 51 (3), 1–31.

Barmpatsalou, K., Cruz, T., Monteiro, E., Simoes, P., 2018b. Current and future trends in mobile device forensics: a survey. ACM Comput. Surv. 51 (3).

Electronics, C.S.S., 2020. OBD2 Explained - A Simple Intro [Online]. Available: https://www.csselectronics.com/screen/page/simple-intro-obd2-explained/language/en. (Accessed April 2021).

Elenkov, N.. Android backup extractor [Online]. Available. https://github.com/nelenkov/android-backup-extractor.

Garfinkel, S., 2007. Anti-forensics: techniques, detection and countermeasures. In: 2nd International Conference on I-Warfare and Security, vol. 20087, pp. 77–84.

Hitchcock, B., Le-Khac, N.-A., Scanlon, M., 2016. Tiered forensic methodology model for Digital Field Triage by non-digital evidence specialists. Digit. Invest. 16, S75–S85.

Horsman, G., 2021. The COLLECTORS ranking scale for 'at-scene'digital device triage. J. Forensic Sci. 66 (1), 179–189.

Jansen, W., Ayers, R., 2007. Guidelines on Cell Phone Forensics, vol. 800. NIST Spec. Publ., pp. 101–800

Kao, D.-Y., Wu, N.-C., Tsai, F., 2019. The governance of digital forensic investigation in law enforcement agencies. In: 2019 21st International Conference on Advanced Communication Technology. ICACT), pp. 61–65.

Lacroix, J., 2017. Vehicular Infotainment Forensics: Collecting Data and Putting it into Perspective. PhD Thesis.

Lukito, N.Y.P., Yulianto, F.A., Jadied, E., 2016. Comparison of data acquisition technique using logical extraction method on Unrooted Android Device. In: 2016 4th International Conference on Information and Communication Technology. ICoICT, pp. 1–6.

Mahajan, A., Dahiya, M.S., Sanghvi, H.P., 2013. Forensic Analysis of Instant Messenger Applications on Android Devices arXiv Prepr. arXiv1304.4915.

Mandal, A.K., Panarotto, F., Cortesi, A., Ferrara, P., Spoto, F., 2019. Static analysis of Android Auto infotainment and on-board diagnostics II apps. Software Pract. Ex. 49 (7), 1131–1161.

Miller, T., 2021. Which OBD2 protocol is supported by my vehicle? [Online]. Available: https://www.obdadvisor.com/obd2-protocol-supported-vehicle/.

Mislan, R.P., Casey, E., Kessler, G.C., 2010. The growing need for on-scene triage of mobile devices. Digit. Invest. 6 (3–4), 112–124.

Nonda. Why is the dashboard data delayed? [Online]. Available. https://nonda.zendesk.com/hc/en-us/articles/360046323211-Why-is-the-dashboard-data-delayed-.

Pankova, T., 2019. WhatsApp forensics: advanced methods of extraction and decryption. DFWRS EU [Online]. Available. https://dfrws.org/presentation/whatsapp-forensics-advanced-methods-of-extraction-and-decryption/.

Park, K.J., Park, J.-M., Kim, E., Cheon, C.G., James, J.I., 2017. Anti-forensic trace detection in digital forensic triage investigations. J. Digit. Forensics, Secur. Law 12 (1), 8.

Rogers, M.K., Goldman, J., Mislan, R., Wedge, T., Debrota, S., 2006. Computer forensics field triage process model. J. Digit. Forensics, Secur. Law 1 (2), 2.

Sim, A.X.A., Sitohang, B., 2014. OBD-II standard car engine diagnostic software development. In: 2014 International Conference on Data and Software Engineering. ICODSE), pp. 1–5.

Singh, A., 2020. How to Root Xiaomi Redmi 9 and Unlock Bootloader (Guide). YtechB [Online]. Available. https://www.ytechb.com/how-to- root-redmi-9-and-unlock-bootloader/.

Srivastava, H., Tapaswi, S., 2015. Logical acquisition and analysis of data from android mobile devices. Inf. Comput. Secur. 23 (5), 450–475.

Thakar, A.A., Kumar, K., Patel, B., 2021. Next generation digital forensic investigation model (NGDFIM)-Enhanced, time reducing and comprehensive framework. In: Journal of Physics: Conference Series, vol. 1767, 12054.

Whelan, C.J., Sammons, J., McManus, B., Fenger, T.W., 2018. Retrieval of infotainment system artifacts from vehicles using iVe. J. Appl. Digit. Evid. 1 (1), 30.

Wijnberg, D., Le-Khac, N.-A., 2021. Identifying interception possibilities for WhatsApp communication. Forensic Sci. Int. Digit. Investig. 38, 301132.

Zaldivar, J., Calafate, C.T., Cano, J.C., Manzoni, P., 2011. Providing accident detection in vehicular networks through OBD-II devices and Android-based smartphones. In: 2011 IEEE 36th Conference on Local Computer Networks, pp. 813–819.

Zhang, X., Baggili, I., Breitinger, F., 2017. Breaking into the vault: privacy, security and forensic analysis of Android vault applications. Comput. Secur. 70, 516–531.