INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

Forensic analysis on Android wear

Sana Shekh
sanashekh12@gmail.com
National Law Institute University, Bhopal, Madhya Pradesh

Dr. Atul Kumar Pandey
atul.pandey@nliu.ac.in
National Law Institute University, Bhopal, Madhya Pradesh

Ankur Rajput
ankurrajput@nliu.ac.in
National Law Institute University, Bhopal, Madhya Pradesh

ABSTRACT

With smartphones floating everywhere in the market, it will not be wrong to say that we are living in a smart era. Nowadays along with Smartphones, the use of wearable devices is also on rising, in which smart watches are topping the charts. Even though small, these devices contain a lot of useful information which can be used as potential evidence in case of cyber-crimes. This paper defines the concept of Android wear along with outlining the basic architecture. It also covers the procedure of data extraction from these smart devices and collection of artifacts even in the absence of mobile phone. The paper explores smart watches as a potential source of evidence along with the crucial information it stores which can be used to solve cybercrimes.

Keywords—*Android wear, Smartwatch forensics, Digital forensic, Internet of Things, Forensic investigation*

1. INTRODUCTION

According to the annual report of Statcounter (2019), the operating system which is currently ruling the market with 75% users is Google's Android. Apart from being at the top in terms of Smartphones, Google entered the market of wearable in the year 2014. It's just been five years, and the Android wearable by Google is already the top choice of many users across the world. The speed with which it is been accepted clearly states that its use is expected to increase in the near future. But the more easily accessible it is, the more it is easier to find crimes in which these smartwatches are involved. Hence there is an emerging need of forensics for these smart wearable devices.

2. ANDROID WEAR

Android Wear, now known as Wear OS, is basically a version of the existing Android OS specifically made for smartwatches and other such wearable devices. The major use of such devices is that just in a glimpse; these devices provide the user with the notifications and various other additional features. "This energetic expansion of miniature computing devices has led to the concept of The Internet of Things (IoT) which can be described as a collection of interconnected and interactive devices which are able to communicate useful real-time information between one another.

The requirements for the Android Wear are:

- A Smartphone with Android version 4.3 or above
- Bluetooth/Wi-Fi connectivity for syncing purposes
- Android Wear APP to keep activities synced on both the ends
- Internet connection

3. NEED FOR SMART WATCH FORENSICS

The birth of these wearable smart watches began back in the 1980s with the invention of Databank CD by Casio which had a lot of amazing features including data storage. So the concept as a whole is not something very new. Android Wear is built on the same concept with slight modifications. But the functionalities and features provided by the Android Wear are far more and advanced as compared to the Databank CD. Therefore, it is necessary to understand at the root level how these devices work and what all data is stored in them so that if it is standalone obtained from the crime scene, the investigator knows what all information can be gathered from such devices.

Also, the number of cybercrimes is increasing day by day. So there is a dire need to understand the forensics of such devices because they might be either used by the criminal to commit a crime or may be used by the victim in synchronization with his/her smart phone. Thus, in either case, it can be a very crucial source of evidence.

4. ARCHITECTURE

The architecture of Android Wear follows the architecture of the Android OS. The kernel of Android Wear is also same as that of Android OS. This device works wirelessly over a Bluetooth/ Wireless Fidelity connection which is established between the Smartphone and the Android Wear. Once this pairing is done, notifications are sent to the watch by the OS to trigger specific actions.

The design guideline document of Google outlines the basic wearable device as shown below:

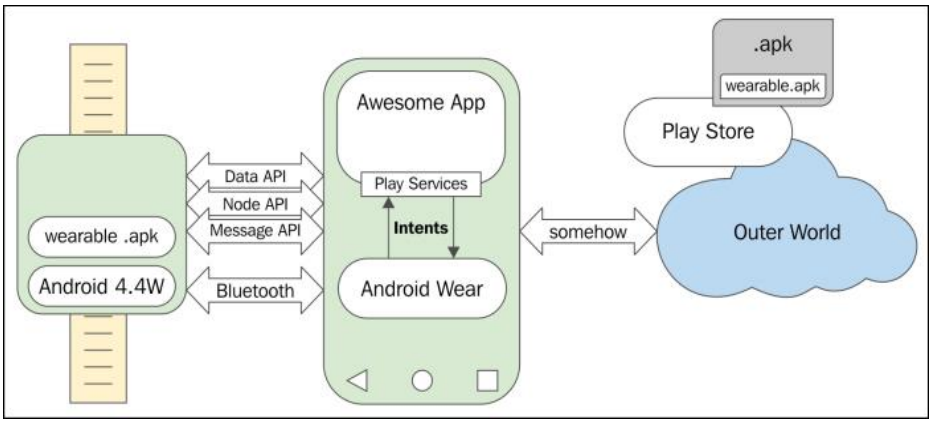

Fig. 1: Structure of android wear [1]

The basic structure of Android Wear is mainly based on three APIs.

- **Data API**: The synchronization of data between the Smartphone and the wearable is possible with the help of data API. The best part about this is, if in case the disconnection between the Smartphone and the wearable happens, the data is automatically transferred once they both are synced again.
- **Node API**: All the connected and disconnected nodes in a wearable network is managed by the node API. A Smartphone is informed when a new node connected with the help of node API only.
- **Message API:** The API calls between the devices is managed by this API. The connected nodes between the wearable device and the handheld device are sent short messages with the help of message API. This provides a one-way communication by sending a small payload.

5. PROCESS FOR SMART WATCH FORENSICS

SmartWatch forensic is quite similar to Android phone forensics because the OS of Android wear belong to the family of Android OS and they both have the same base of Linux.

5.1 Prerequisites

5.1.1 Phone

- Enable the developer option by going to about>tapping build number multiple times.
- Enable debugging by clicking on developer options.

5.1.2 Computer

- Download and install the Android SDK on the machine.
- Download the universal ADB drivers and reboot the machine.

5.1.3 Watch

- Enable the developer option by going to about>tapping build number multiple times.
- Enable ADB debugging.
- Debug over Bluetooth.

5.2 Rooting the watch

- Before rooting, download the latest version of adb and fast boot binaries on your computer and set path variable for them.
- Download the rooting boot image .[2]
- Enable the developer option and then enable ADB debugging.
- Use the command ‘adb reboot boot loader’ to boot into fastboot boot loader mode. (Make sure that the boot loader is unlocked).
- After this, the boot image is booted with the command ‘adb boot SWR50-root boot-for-LWX48P-byMR.img. [2]
- This will start the task of booting the image. When the rooting boot image completed its tasks it will automatically reboot into the regular Android system.
- To check if the device is rooted, type ‘adb shell’ and then execute the command su (super user). If the device is rooted, it will show ‘root@tetra:/ #’.
- This shows that the device is successfully rooted.

5.3 Procedure

5.3.1 If the Mobile phone is not present

- Connect the Wear watch to the computer via USB cable.

- Open up the command prompt and type in ***'adb devices'*** and it will show the device connected. It will only show the connected device if it was factory reset. If it wasn't factory reset, the RSA authentication key will be sent to the mobile phone device which is not available.
- Once the device is listed, one can proceed.
  Ex- F:\Androidid\sdk\platform-tools>adb devices
  List of devices attached
  14432D2BF44FFF2 device
- Since the thumb rule of investigation states that the integrity of the evidence must not be compromised, the image of the device under investigation must be created so that the original evidence remains intact. Calculate the hash before and after imaging so that the integrity may be proved. Further, to avoid any accidental modifications during the imaging process, the evidence device should be in write protected mode.
- Execute the command ***'df'*** in the ADB shell and the directories will get listed
  Ex-
  /boot
  /system
  /recovery
  /data
  /cache
  /misc
  /sdcard
- Chose the directories which will have data for the purpose of imaging which includes /system, /data, / cache.
- To display all the memory blocks, type **ls -al /proc/partition/**. This will show all the partitions.
- To make partition specific image, execute the command in the format-
  dd if=/mountingpointpartition of=/destination blocksize
  Ex-
  dd if= /dev/block/mmcblk0p29 of=/home/user/boot.img bs=1M

This shows making the image of boot partition of the block device.
Once the image is made, it is ready for analysis.

5.3.2 If the Mobile phone is present

After following the prerequisites for the phone, follow the below steps-

- Enable Bluetooth on the phone and then go to the watch's setting>Bluetooth devices>pair.
- In the android wear app, enable 'Debug over USB'.
- In the command line type :
  adb forward TCP: 4444 local abstract: /adb-hub (This sets up the port)
  adb connect localhost: 4444 (This connects you to the phone via the port)
- The phone will give a prompt to give the computer to connect via ADB to the watch. Click on always allow for this computer.
- The android wear app will show host and target as connected.
- Open up the command prompt and type in ***'adb devices'*** and it will show the device connected. You can see two devices - your phone and a bridged port for the watch.
- Follow the steps' to 'i' from the "if the mobile is not present" as it is. The image will be ready for the analysis.

6. ANALYSIS

The analysis of the image is the next major step. This analysis can be done with the help of various tools like FTK imager, Scalpel etc. The main partitions which have to be focused for the purpose of forensics are:

6.1 System

This memory block consists of a variety of information. The information stored here is specific to the device system. The most important information which can be obtained is media files, bin, file system etc. It can also provide information about frameworks, default fonts etc.

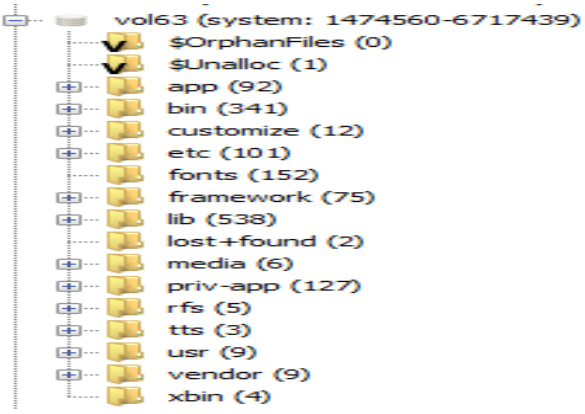

Fig. 2: Contents of system partition

6.2 Data

Even though smart watches do not store a lot of data, but whatever data is stored in the device is stored at this location. The data is stored here in database format and DB browser for SQLite can be used to read the '.db' files. It stores data of application of smart phones to which the smart watch has been synced with. It stores installed apps, messages, user settings etc.

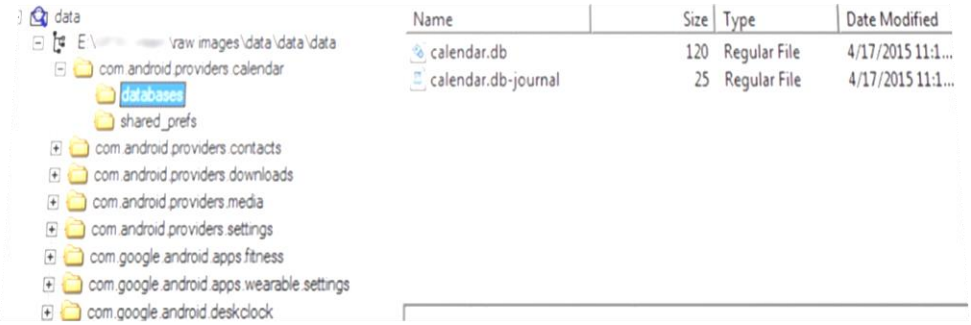

Fig. 3: Contents of data partition

6.3 Cache

Just like the cache in computer systems, cache here also serves the same purpose of quick access by storing recently accessed applications. The cache data of an application is stored in this partition. This contains a variety of information like recovery information and also information related to the last install, last log etc. which are very crucial in case of tracking a person.

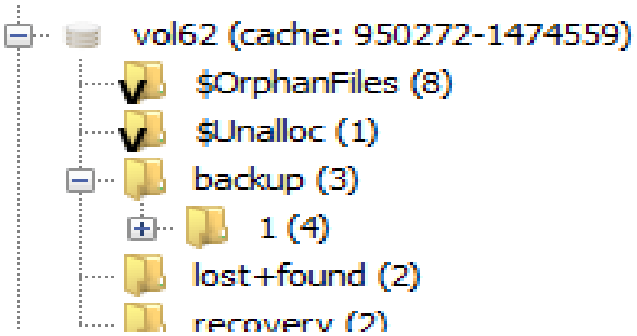

Fig. 4: Contents of cache partition

7. ARTIFACTS OF SONY SMARTWATCH3 (MODEL # SW50)

- Paired device information

/data/misc folder

- Connected Bluetooth devices, the device id and Mac address

bt_config.old

- Voice commands

/data/data/com.google.android.gms

- Command triggered to do tasks

voice_action_title

- Dropbox information

/data/system/dropbox

- Recent task

/data/system/recent task

- Wearcamera information

/data/media

- Recent images

/data/system/recent images

- Bluetooth packets can be captured using any network/Bluetooth packet capturing software.
- Logcat command can be used to view logs from different applications.
- Notification from the various app can be found in google gsm services>name of the application.

8. CONCLUSION

Belonging to the Android family made the Android Wear Smart watch forensics examination a lot more familiar process. Even though the smart watches do not store the complete information, it still stores substantial information in scattered form. This information can be linked together to help in the investigation process. There is still a lot of void in the field of wearable device forensics which needs to be filled as soon as possible keeping in mind the rising cybercrimes.

9. REFERENCES

[1] Available at <https://subscription.packtpub.com/book/application_development /9781785280153/ 1/ch01lvl1sec09/ understanding-the-android-wear-architecture>

[2] Available at <https://drive.google.com/open?id=0BzOF0XGjhOopWHozY0NnV2E1Z2c>, Accessed on 13/4/19
[3] https://www.dataforensics.org/smartwatch-forensics/
[4] https://subscription.packtpub.com/book/application_development/9781785280153/1/ch01lvl1sec11/building-a-simple-android-wearable-application
[5] https://docs.microsoft.com/en-us/xamarin/android/wear/get-started/intro-to-wear
[6] https://www.ijser.org/researchpaper/Analysis-of-Android-Smart-Watch-Artifacts.pdf
[7] https://www.researchgate.net/publication/283734724_Watch_What_You_Wear_Preliminary_Forensic_Analysis_of_Smart_Watches
[8] https://www.researchgate.net/publication/283734724_Watch_What_You_Wear_Preliminary_Forensic_Analysis_of_Smart_Watches
[9] https://app.box.com/s/al8pucdphtsjjafas4dx

