Investigation of security on Chinese smartwatches

Renee Witsenburg
rwitsenburg@os3.nl

Kasper van Brakel
kbrakel@os3.nl

Abstract—This study investigated the information stored on smartwatches and how to retrieve this information from three Chinese smartwatches: the Xiaomi Amazfit Bip, the Lemfo KW18, and the Lemfo LEM8. An estimation of smartwatch usage within a business environment has been made, including the risks these use cases bring. This depends on the functionalities of the watches, the usage of the watch, and what data is stored on them. However, data from contacts and Whatsapp messages synchronised from the mobile phone can be retrieved from the flash memory of the Lemfo smartwatches. Through USB connection a flash memory extraction can be performed with Flash tool and with a Bluetooth connection it was possible to send fake notifications to the Lemfo KW18 and the Amazfit Bip. The investigated smartwatches have limited security measures in place, root permission is not required in order to extract data through an USB connection. The Bluetooth Low Energy stack ensures that there can only be one active connection with a device simultaneously, which limited the interception possibilities. Awareness on the possibility that information can be stored on a smartwatch is needed. Some smartwatches have security functions, like a pin code or a remote wipe option. According to the Dutch National Security Center, it is best practise to use these options.

Index Terms—Smartwatches, Wear OS, Android Wear, Bluetooth, Xiaomi Amazfit, Lemfo

1 Introduction

Wearables are gaining popularity on the market according to Statista [8]. Users want to monitor their daily activity patterns and get used to a more healthy lifestyle. The data of these wearables is stored on the mobile telephone the device is paired with. This data may consist of age, gender, weight, heart rate, geoposition, activity periods, and sleep habits. In other words: personal information that could be marked as sensitive. Often, this information is not only stored locally on the mobile telephone, but also stored on an online account on remote servers of the vendor. Analyses and new training schedules can be made based on gathered data. These wearables get more functions over time, and notifications from other applications can be received on these devices. This study will analyse smartwatches produced by Chinese vendors. The definition of a smartwatch according to this study will be: a wristband with sensors. Sensor information from the wristband is sent to a mobile telephone. Furthermore, notifications from the mobile telephone are sent to the wristband.

Personal information from a smartwatch, like GPS coordinates, could be linked to the location where the owner of the watch lives or works. This is not meaningless information, but it is rather sensitive information. Secret training locations of the military, war zones, or the locations of the secret service were exposed due to smartwatches [9]. On some smartwatches, incoming notifications from apps on the mobile phone can be displayed. The context of the notification is determined by whether sensitive information is exposed. It is becoming more common to wear smartwatches and other wearables in a business environment. Companies need to have a policy for using wearables and the content that is displayed on them. The Dutch National Security Center has a paper with security guidelines on mobile devices that can be applied on smartwatches [5].

This study is about the significance of the information stored on the smartwatch, the incoming information on a smartwatch, and the information on a smartwatch after loss or theft. The focus is on smartwatches in a business environment. Smartwatches that potentially contain business sensitive information could also be worn when not at work, which could increase the risk of data leakage.

Section two contains the research questions of this study. Section three presents related research about the topic of smartwatches. This will include the research about sportwatches and smartwatches as well as the wireless connections that smartwatches can make. Section four outlines the methods that are used to gather information from the smartwatches and the attack scenarios will be explained and described in detail. Section five contains the results of the experiments. Section six contains the discussion of the results and work done. Section seven presents the conclusion based on the results found. Lastly, section eight outlines future work.

2 RESEARCH QUESTIONS

For this study, a forensic investigation on three smartwatches will be performed. According to Kevin Olivieri from IBM, the built-in security for smartwatches is considered to be still in its infancy [21]. During this study, both technical and human factors are investigated, including the way people are using wearables in a business environment. It could be the case that sensitive company data is stored on these wearables, which means that a data leak could occur when a smartwatch is lost or stolen. Furthermore, the possible ways to intercept data that is sent between the smartwatch and mobile phone and vice-versa will be investigated.

The main research question of this project is:

When smartwatches are used in a business environment, what potential information leakage risks are encountered and what solutions are available?

In order to answer the main research question, the following sub-questions are defined:

- For what purposes are smartwatches used in a business environment?
- Which connections can be made with the smartwatch?
- What security measures are in place?
- What data is stored on the smartwatch?
- Is it possible to tamper with, read or intercept this data?

3 RELATED WORK

Previous research on smartwatches and the connections they support has already been done. In this section related work about sportwatches and smartwatches, extracting data, and wireless attacks will be discussed.

3.1 Smartwatch architecture

Security researchers performed analyses on smartwatches and information stored on the device, in the mobile application and, when available, in the cloud. In 2018 Classen et al. performed a study on the anatomy of the Fitbit tracking system [12]. They studied the device, the mobile application on the phone, the communication between these entities, and the infrastructure of the cloud environment where information was stored. They found multiple vulnerabilities, and they concluded ”that the original Fitbit architecture was not designed with security in mind from the start” and ”Security was added while products were already shipped to end users, resulting in trackers that are potentially still compromised.” [12]

The data on sportwatches with the Advanced and Adaptive Network Technology+ (ANT+) protocol can be extracted with the proper forensic tooling, as proved by Kessel and Laan [25]. Kessel and Laan wrote a paper about sportwatches that made a connection with smartphones through the ANT+ protocol. They studied the impact of data modification. The researchers found that all of the data is stored locally on the device and the researchers were able to read the data from the device. However, tampering with the data was not possible, because write permission via the the ANT-FS protocol was disabled. They concluded that the data on the watches studied is reasonably secure.

In 2015, Dreijer and Houtenbos studied the Android Wear platform and attempted to discover how to find forensic data on Android smartwatches with existing mobile forensic tooling [13]. In the same year, little research has been done into Android Wear or the now called Wear OS. The developer version of Android Wear was released in March 2014. However, Android Wear or Wear OS is based on the regular Android OS, and it is possible to do forensic research and extract data through the Android Debug Bridge (ADB). The SANS institute offers training’s and tutorials on smartphone forensics and pentesting [10]. With one of these tutorials [4], the researchers could extract data from the smartwatch and read potentially private information from WhatsApp and Gmail. However, they could not retrieve data from a live running device.

3.2 Extracting forensic data

A considerable amount of research on how to carry out a forensic investigation on mobile devices has already been done. For example, the SANS institute provides courses on how to extract data from mobile telephones like iPhones and Android phones [10]. However, because of the wide variation of mobile phones, there are many differences in hardware. This makes extracting data from mobile devices hard. In 2015 Joe Kong [20] evaluated three generic tools to extract forensic data from a Lenovo A850 smartphone. He used this phone for his study because the phone is equipped with a MediaTek chip. The Android phones with MediaTek chips are often used in crime cases because they are cheap, but have a high price /performance ratio of the CPU. However, the more specific forensic tools for MediaTek based phones only handle a limited number of models, and the newer models are not supported. His study concluded that more generic forensic tooling could also extract data from the phone. Mengfei He discovered in 2012 that the data stored on the mobile devices from Shanzhai, which are also based on the MediaTek chip, could be fully extracted, and the events and data that are stored on the phone are not encrypted [17].

3.3 Wireless communication

Smartwatches use wireless technology to connect with a mobile phone. Most smartwatches are connected with a form of Bluetooth [22]. Bluetooth is an open standard and free to use on short ranges. This standard makes use of a Bluetooth stack with transport protocols and middleware protocols.

Since smartwatches are small devices, they have a relatively small battery. Therefore Bluetooth Low Energy technology is often used. In 2012, an overview and evaluation of Bluetooth Low Energy (BLE) was published by Carles Gomez et al. [15]. They described the differences between previous implemented Bluetooth versions and the Low Engergy (LE) variant. BLE is developed by the Bluetooth Special Interest Group (SIG) for short range communication. BLE is used for healthcare, consumer electronics, smart energy, and security.

The Bluetooth stack can be split into two parts, as can be seen in figure 1. The controller holds the physical layer and the link layer. The host holds the upper layer functionality, the Logical Link Control and Adaptation Protocol (L2CAP), Attribute Profile (ATT), Generic Attribute Profile (GATT), Security Manager Protocol (SMP), and General Access Protocol (GAP). The communication between the controller and the host is managed by the Host Controller Interface (HCI).

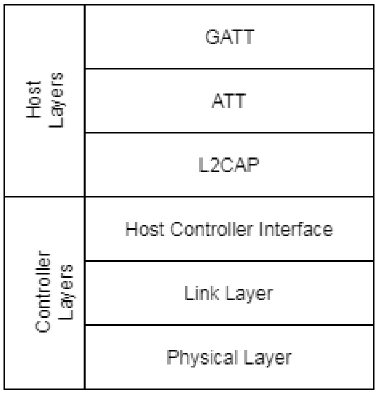


Fig. 1. Bluetooth stack

With BLE, a device broadcasts data through advertising packets. It sends information about what kind of device it is and that it is available to pair with other devices. When a device wants to set up a connection with the advertiser, it sends a connection request to the advertiser. When this point-to-point connection is established, both devices can communicate by using physical data channels. In this connection, the master and slave are defined. Most often, the BLE-device is put in the slave role. This network connection is called a *piconet*. A BLE-device, like a smartwatch, can only belong to one piconet.

For Bluetooth Low Energy (BLE), a Low Energy implementation for L2CAP is made. This protocol is used for multiplexing the data of higher level protocols. The ATT protocol is used to let a client and a server communicate. The client can send requests, but the server can also send unsolicited messages that for example contain notifications. The GATT framework can be used on top of the ATT protocol. This framework uses ATT for discovering services. Values and properties can be sent via the GATT framework.

In the whitepaper ”Gattacking Bluetooth smart devices” [18], a clear explanation of the security features of the BLE technology is given: ”The Bluetooth Core Specification provides several features to cover the encryption, trust, data integrity and privacy of the users data.” [23]

To use encryption between two devices, a pairing process takes place. There are a three methods to set a long term key. These options are Just works, Passkey Entry, and Out of Band. The method chosen depends on the available options of the device. According to the Bluetooth SIG ”Just Works and Passkey Entry do not provide any passive eavesdropping protection” [23]. As discovered by Jasek, a significant number of smart devices did not implement the security features in BLE provided by the Bluetooth SIG. Some vendors do not consider the risk or possibility of the transmission being intercepted, and others prioritise usability rather than security. [18]

Less common are smartwatches that are equipped with WiFi or 4G. On some smartwatches these options are available though. In WiFi connectivity, vulnerabilities and exploits are already found and described. On these smartwatches, WiFi is another attack vector. The use of the 4G LTE network is another option next to WiFi. Moreover, vulnerabilities in the 4G LTE network have been discovered in this network as well [19].

3.4 Bluetooth attacks

Since the way Bluetooth works, as well as the way its protocols are implemented are public information, various attacks can be thought of. A Bluetooth attack could steal personal information, such as photo's, text messages, or calendar schedules. Other possible Bluetooth attacks that could be thought of, are eavesdropping or gaining full control of the devise. Furthermore, Denial-of-Service (DoS) attacks could be thought of. DoS attacks may paralyze the device or drain the battery.

In the article *Security Threats from Bluetooth Technology*, Hassan et al. write an overview of possible Bluetooth attacks. "Attacks which follow similar method [sic] of penetration or leave the same effect on the victim are grouped under one single title." [16] as can be seen in figure 2. Attacks can also be grouped based on severity for the victim.

Jasek wrote a whitepaper about attacking low energy devices [18]. The attack scenarios were based on real-life vulnerabilities in the BLE technology, but his focus was on the GATT layer of the BLE stack. When the smart device is broadcasting advertisements, it is possible to jam and DoS the device. These broadcast packets can also be used to change the status of an application. In home automation, this could lead to attacks with a minor impact like being unable to turn on your (Bluetooth connected) lights. However, there are also attacks with a greater impact. In the newest Android versions, it is possible to keep the phone unlocked when the Bluetooth device is nearby. When the MAC-address from the LE device is spoofed it is possible to keep the phone in an unlocked state, which can have severe consequences. Jasek developed the GATTacker tool to attack LE devices in various ways. Attacks like DoS, spoofing, passive and active transmission interception, and abuse of the device are available. The tool provides all the possible attacks that are described in the whitepaper that he wrote.

4 METHODS

For this study, three Chinese smartwatches were investigated.

- LEMFO KW18
- LEMFO LEM8
- Xiaomi Amazfit Bip

In order to determine for which purposes smartwatches are used in a business environment a questionnaire was composed. The questions from the questionnaire are inspired by the questions from the measurement model of previous research that investigates the role of usefulness and visibility in smartwatch adoption [11]. The questions can be found in Appendix Table 3.

4.1 Obtaining data

To determine the connectivity possibilities of the smartwatches, the specifications of the vendors were analysed. Besides that, the smartwatches were analysed on connections as USB, Bluetooth, WiFi and 4G LTE. The devices was not disassembled for this research.

An aspect of this research was to investigate how the data of the smartwatches can be retrieved. To gather the data on the smartwatch multiple methods were used. In the case of loss or theft of the device, physical access is possible. The device can be connected to a computer and forensic tooling can be used. According to Joe Kong, a tool called Flash tool can be used to extract data from devices powered by a MediaTek chip [20]. In order to extract the data from such devices Flash Tool has to be provided with a scatter-file, which contains the start and end addresses of the memory blocks that exist on the target device. The user can either choose to write data to these memory locations or choose to read the data that is located on the specified memory locations and store it on the host computer.

To investigate whether it is possible to extract data from the smartwatches and to investigate if the data is encrypted, a text file that contains the string 'thisisatest' was stored on the local storage of the device. A data dump will be extracted from the device and the strings command `strings -a --radix=x FILE_01_mtk | grep thisisatest` in Linux was executed on the data dump. When the string 'thisisatest' exists it means that the data was successfully extracted from the smartwatch.

4.2 Wireless attacking the device

To investigate how the smartwatches can be read out and manipulated while wearing them, several Bluetooth attacks were executed. Hcitool was used to detect Bluetooth and Bluetooth Low Energy devices. Devices that were connected to an Android phone, logged their data to the Snoop log files on the telephone. These snoop Bluetooth HCI files were

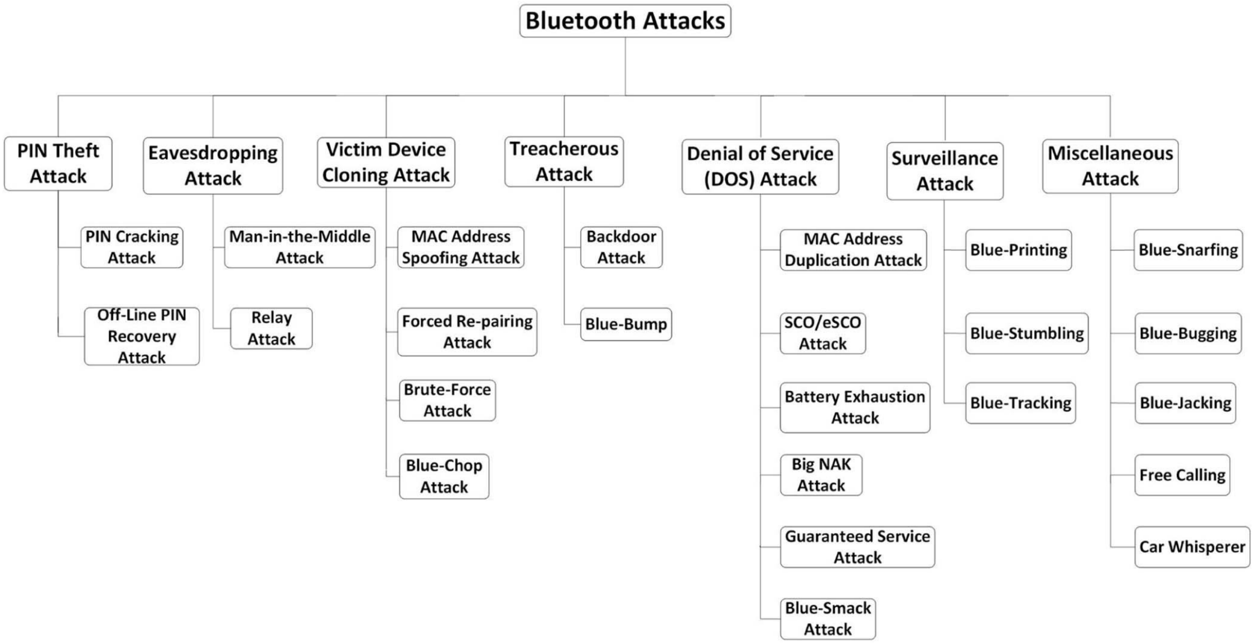


Fig. 2. Categories of Bluetooth attacks [16]

extracted with ADB and analysed with tools like Wireshark.

With the Ubertooth framework, Bluetooth traffic was monitored [2]. It could be interesting to view the communication and the connections that are established between the target device and, for example, a mobile phone. The Ubertooth framework can also be used to analyse whether the wireless traffic is encrypted on the Bluetooth stack or if there are any other security measures in place.

The mobile app nRF Connect can display the UUID's of the services that are running on LE smart devices. The services can be used in the app. It is possible to create macros in the Android version of the app [6] to save functions of the services, like simulating the call notification on a smartwatch.

Andrey Nikishaev wrote a blog about a tool to read out the live data from his Xiaomi MiBand 2 [7]. Because the MiBand 2 has some similarities with the Xiaomi Amazfit Bip, this tool was used as example to extract the live data from the Amazfit and to send notifications.

To perform some Bluetooth attacks, Gattack was used. Gattack gathers information about services and characteristics of the device. It clones the device to Man-In-The-Middle the connection. It should be possible to intercept and modify the transmitted requests and responses. To spoof the Bluetooth MAC-address of the devices, a Raspberry Pi 3 was used. This device has a compatible onboard Bluetooth chip to perform this action.

5 Results

The questionnaire was sent to people that are working in a security department of a large international cyber security company or at a cyber security organisation of the Dutch government. Half of the respondents wear a smartwatch at work. Most of them wear an Apple watch or a Fitbit. More than half of the respondents that wear a smartwatch assume that business information could be stored on their smartwatch. The information that they receive on their smartwatch mostly consists of e-mails, agenda notifications, and text messages. Fourteen of the respondents that wear a smartwatch also use a pin code on their watch. However, it needs to be mentioned that not all smartwatches offer this function. Less than fifty percent of the respondents who wear a smartwatch have taken precautions in case of loss of the smartwatch.

During this study three smartwatches from Chinese manufacturers, with similar functions as can be found in the smartwatches from the respondents, were investigated. The smartwatches which have been studied have different specifications. A graphical overview of these specifications can be found in table 1.

5.1 Physical access

In the case of a smartwatch getting lost or stolen, physical access to the smartwatch is granted. When a pin code is used on the device, system information or user data cannot be accessed without extraction tools.

Model	Xiaomi Amazfit Bip	Lemfo KW18	Lemfo LEM8
OS	proprietary OS	MRE (Maui Runtime Environment)	Android 7.1.1
CPU	ADI Dual core 1.2GHz	MT2502C Dual core 1.2GHz	MT6739WA ARM Cortex-A53 Quad core 1.5GHz (64bit)
RAM	128 MB	64 MB	2 GB
Internal memory	N/A	128 MB	16 GB flash memory
SDcard slot	N/A	Max 16GB	N/A
USB	Only charging	Charging & data transfer	Charging & data transfer
Bluetooth	4.0 BLE	4.0	4.0 BLE
Wi-Fi	N/A	N/A	802.11 a/b/g/n
Sim card slot	N/A	Nano SIM (2G)	Nano SIM (4G LTE)
GPS	Glonass	No	Glonass, Beidou
Microphone	N/A	Yes	Yes
Speaker	N/A	Yes	Yes
Smartphone APP	MiFit 3.0	Fundo Wear	WiiWatch2
Heart rate sensor	PPG optical	Yes	Yes
Magnetometer	Yes	N/A	N/A
Accerelometer	Yes	N/A	Yes
Pedometer	Yes	Yes	Yes
Gyroscope	Yes	N/A	Yes
Light sensor	Yes	N/A	Yes
Battery	190 mAh	340 mAh	580 mAh

TABLE 1
Specifications overview of the investigated smartwatches.

5.1.1 Xiaomi Amazfit Bip

The Amazfit Bip is a sportwatch that offers functions such as: step counting, measuring heart rate, counting calories and sitting time. In addition, it can receive notifications. A pin code cannot be set on the device. The USB connector is only used to charge the device and therefore no data extraction through USB cable could be performed.

5.1.2 Lemfo KW18

The Lemfo KW18 is a smartwatch on which a pin code cannot be set. This makes it possible to read user data from the watch itself. When connected through USB, data extraction can be performed on a computer. Extracting data from the smartwatch can be done with Flash tool. When the start address, 0x0000000, and the end address, 0x01000000, are provided to the Flash tool, the data can be retrieved. After performing the data dump, a single data file is stored on the host's computer as result.

It was discovered that the operating system on the device is most likely a MRE (MAUI Runtime Environment). MRE is developed by MediaTek and it was built for smart devices just like Android [24]. When the combination of the characters ”* # 9966 * # ” is dialed in the dial app, the full device information is displayed, including MRE_version = 3100, which suggests that the KW18 is indeed a MRE based device.

```
$ strings -a -t x backup4.img | grep
    thisisatest
 e78400 thisisatest
```

Listing 1. KW18 examining data dump with strings command.

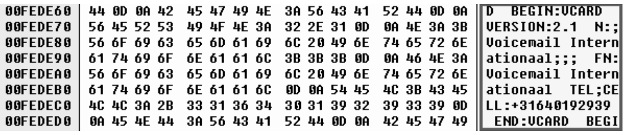
As can be seen in listing 1 the test file that was placed on the watch is successfully retrieved with Flash tool with the method described above. The whole flash drive is currently stored in one file.

When analysing the data dump file with the command `file` on a Linux system, ”data” is returned as a result. In order to get more comprehensive information about the data structure, the tool Binwalk was used and the data was inspected with a hex editor. Table 2 gives an overview of the structure that was identified:

Part	Start(hex)	End(hex)
SF_BOOT	00000000	000001F0
BRLYT	00000200	000007F0
int_bootloader	00000800	000028C0
Padding	000028D0	00005FF0
ext_bootloader	00006000	0000FB90
padding	0000FBA0	0001FFF0
FILE_01_mtk	00020000	00BE5000
User data	00BE5010	00FFFFF0

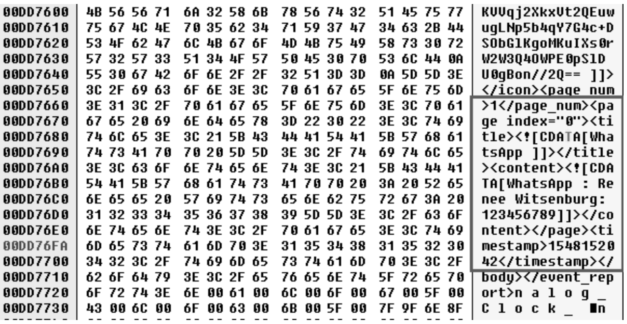
TABLE 2
A rough overview of the data structure from the data dump of the KW18.

The int_bootloader, ext_bootloader and FILE_01_mtk were recognised when comparing the data dump in the hex editor with the firmware files found on XDA developers [14]. The remaining part, which is referred to as user data in table 2, containing mobile phone contacts and Whatsapp notifications, can be found in plain text. This can be seen in figure 3 and figure 4. It is possible to extract parts of the data dump with the `dd` command on a Linux system. In this study, the structure of the user data partition is not further investigated since the data could already be extracted.



```
00FEDE60  44 0D 0A 42  45 47 49 4E  3A 56 43 41  52 44 0D 0A  D  BEGIN:VCARD
00FEDE70  56 45 52 53  49 4F 4E 3A  32 2E 31 0D  0A 4E 3A 3B  VERSION:2.1  N:;
00FEDE80  56 6F 69 63  65 6D 61 69  6C 20 49 6E  74 65 72 6E  Voicemail Intern
00FEDE90  61 74 69 6F  6E 61 61 6C  3B 3B 3B 0D  0A 46 4E 3A  ationaal;;;  FN:
00FEDEA0  56 6F 69 63  65 6D 61 69  6C 20 49 6E  74 65 72 6E  Voicemail Intern
00FEDEB0  61 74 69 6F  6E 61 61 6C  0D 0A 54 45  4C 3B 43 45  ationaal  TEL;CE
00FEDEC0  4C 4C 3A 2B  33 31 36 34  30 31 39 32  39 33 39 0D  LL:+31640192939
00FEDED0  0A 45 4E 44  3A 56 43 41  52 44 0D 0A  42 45 47 49   END:VCARD  BEGI
```

Fig. 3. KW18 vcard example



```
00DD7600  4B 56 56 71  6A 32 58 6B  78 56 74 32  51 45 75 77  KVVqj2XkxVt2QEuw
00DD7610  75 67 4C 4E  70 35 62 34  71 59 37 47  34 63 2B 44  ugLNp5b4qY7G4c+D
00DD7620  53 4F 62 47  6C 4B 67 6F  4D 4B 75 49  58 73 30 72  SObGlKgoMKuIXs0r
00DD7630  57 32 57 33  51 34 4F 57  50 45 30 70  53 6C 44 0A  W2W3Q4OWPE0pSlD
00DD7640  55 30 67 42  6F 6E 2F 2F  32 51 3D 3D  0A 5D 5D 3E  U0gBon//2Q== ]]>
00DD7650  3C 2F 69 63  6F 6E 3E 3C  70 61 67 65  5F 6E 75 6D  </icon><page_num
00DD7660  3E 31 3C 2F  70 61 67 65  5F 6E 75 6D  3E 3C 70 61  >1</page_num><pa
00DD7670  67 65 20 69  6E 64 65 78  3D 22 30 22  3E 3C 74 69  ge index="0"><ti
00DD7680  74 6C 65 3E  3C 21 5B 43  44 41 54 41  5B 57 68 61  tle><![CDATA[Wha
00DD7690  74 73 41 70  70 20 5D 5D  3E 3C 2F 74  69 74 6C 65  tsApp ]]></title
00DD76A0  3E 3C 63 6F  6E 74 65 6E  74 3E 3C 21  5B 43 44 41  ><content><![CDA
00DD76B0  54 41 5B 57  68 61 74 73  41 70 70 20  3A 20 52 65  TA[WhatsApp : Re
00DD76C0  6E 65 65 20  57 69 74 73  65 6E 62 75  72 67 3A 20  nee Witsenburg:
00DD76D0  31 32 33 34  35 36 37 38  39 5D 5D 3E  3C 2F 63 6F  123456789]]></co
00DD76E0  6E 74 65 6E  74 3E 3C 2F  70 61 67 65  3E 3C 74 69  ntent></page><ti
00DD76FA  6D 65 73 74  61 6D 70 3E  31 35 34 38  31 35 32 30  mestamp>15481520
00DD7700  34 32 3C 2F  74 69 6D 65  73 74 61 6D  70 3E 3C 2F  42</timestamp></
00DD7710  62 6F 64 79  3E 3C 2F 65  76 65 6E 74  5F 72 65 70  body></event_rep
00DD7720  6F 72 74 3E  6E 00 61 00  6C 00 6F 00  67 00 5F 00  ort>n a l o g _
00DD7730  43 00 6C 00  6F 00 63 00  6B 00 5F 00  7F 9F 6E 8F  C l o c k _  ■n
```

Fig. 4. KW18 whatsapp notification

5.1.3 Lemfo LEM8

On the Lemfo LEM8 there is no option to set a pin code. When the smartwatch is stolen or lost, it is possible to read the user data on the watch itself. The Lemfo LEM8 is a smartwatch running on the Android 7.1.1 operating system. This means that the data on the watch will be divided over the known Android partitions: /boot, /system, /recovery, /cache, and /userdata. More details can be found in the Appendices section in table 4.

When connected with USB it is possible to connect through Android Debug Bridge (ADB) in order to open a shell on the smartwatch. Without root access, it is possible to read out the partition table with the command `cat /proc/partitions`. All the partitions that are visible for a regular user are displayed with the following parameters:

- **major:** major indicates on which storage device the partition is located.
- **minor:** minor value indicates the order the partitions are located on the drive.
- **#blocks:** the size of the partition in number of blocks. The block size is 1024 bytes for the Lemfo LEM8.
- **name:** the name of the partition.

```
major minor #blocks name

 254 0 982220 zram0
 179 0 15267840 mmcblk0
 179 1 1024 mmcblk0p1
 179 2 24576 mmcblk0p2
 179 3 512 mmcblk0p3
 179 4 20480 mmcblk0p4
...
```

Listing 2. Partial output of /proc/partitions

With the output of the command `cat /proc/devices` it is possible to identify the major integer with the corresponding device. The major value 179 is linked to the mmc, which is the flash drive of the Lemfo LEM8. The minor value ”0” represents the complete flash drive while the values 1, 2, 3 etc. represent in which order the partitions are stored on the watch. Meaning that the starting address of the partition with, minor value 2, combined with the length of partition 2, is the beginning value of the partition with minor value 3.

```
Character devices:
...
254 BOOT

Block devices:
259 blkext
  7 loop
...
134 sd
135 sd
179 mmc
253 device-mapper
254 zram
```

Listing 3. Partial output of /proc/devices

Mmcblk0 represents the whole flash disk and therefore contains all the data that is stored on the watch’s flash storage. As can been seen in listing 2, mmcblk0 consists of 15267840 blocks. In order to determine the size of a partition, the number of blocks from the partition overview file have to be multiplied with the block size of the device. The Lemfo LEM8 has a block size of 1024 bytes, resulting in the total size of mmckl0 15,634,268,160 bytes and 3A3E00000 in hexadecimal notation. This memory address can be used to retrieve the data from the smartwatch with Flash tool.

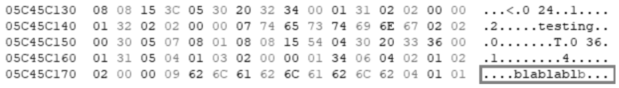
After copying the test file with the name ”thisisatest” to the Lemfo LEM8 smartwatch, Flash tool is provided with 0x0 as starting address and 0x3A3E00000 as ending address, because it was determined that this memory address represents the whole flash disk. Listing 4 displays the result when the data dump is analysed with the strings command.

```
strings -a -t x backup_whole.img | grep
    thisisatest
d02fa19c thisisatest.txt
d802b7ea thisisatest
d802c2cd /storage/emulated/0/thisisatest
    .txt
d802c2fb text/
    plainthisisatest13894445970
d802c31e thisisatest.txt
d802d7cc /storage/emulated/0/thisisatest
    .txt
d802e7e4 /storage/emulated/0/thisisatest
    .txt
d802f315 /storage/emulated/0/thisisatest
    .txt
d802f343 text/
    plainthisisatest13894445970
d802f366 thisisatest.txt
35f20419c thisisatest.txt
35f23d000 thisisatest
```

Listing 4. LEM8 examining data dump with strings command.

As can be seen in listing 4, the test file which was placed on the watch is successfully retrieved with Flash tool using the method described above. Currently, the whole flash drive is stored in one file. Since the existing partitions and their sizes are already known, a scatter-file can be composed in order to read the flash memory of the LEM8 and store each partition in an individual data dump file. The user data partition is an EXT4 file system. This partition can be mounted and then explored. None of the data which was found in this partition was encrypted.

Since the data was successfully extracted from the device, one could think of a physical attack on the device in order to tamper with the data. To perform this attack, in the hex editor some changes were made on the extracted data. As a result, of making the changes that are shown in figure 5, the image file became corrupted and it was therefore not possible to write the tampered data back to the watch.



```
05C45C130   08 08 15 3C 05 30 20 32 34 00 01 31 02 02 00 00   ...<.0 24..1....
05C45C140   01 32 02 02 00 00 07 74 65 73 74 69 6E 67 02 02   .2.....testing..
05C45C150   00 30 05 07 08 01 08 08 15 54 04 30 20 33 36 00   .0.......T.0 36.
05C45C160   01 31 05 04 01 03 02 00 00 01 34 06 04 02 01 02   .1........4.....
05C45C170   02 00 00 09 62 6C 61 62 6C 61 62 6C 62 04 01 01   ....blablablb...
```

Fig. 5. Tampering with the data on the LEM8.

5.2 Bluetooth access

When physical access is not possible, because the owner is wearing the smartwatch, another way to gather the data on the watch is required. Since the three smartwatches investigated have a Bluetooth chip, the Bluetooth attack vector will be investigated.

5.2.1 Xiaomi Amazfit Bip

During the study of the smartwatches, it could be observed that the Amazfit Bip could connect only to one device at a time over Bluetooth. It was discovered that this is a specific characteristic of the Bluetooth Low Energy technology. To use Bluetooth as an attack vector, Hcitool was used, as well as the btsnoop log from the phone in combination with Wireshark, Ubertooth, the nRF connect Android app, and Gattacker.

With Hcitool it is possible to scan and connect with Bluetooth and Bluetooth Low Energy devices. The Amazfit Bip disappeared from the scanning list when a connection with another device was made.

To gather more information about the device, the services that where running, and the authentication process, a Bluetooth snooplog was made and stored on an Android phone. To analyse this logfile, it was opened in Wireshark. In Wireshark it could be seen how the device sends advertising packets, shows services and characteristics, goes through an authentication process with the phone, and updates the A-GPS.

An experiment was done with the Ubertooth framework to sniff and intercept the connection between the watch and the phone. Unfortunately, although the Ubertooth framework claims that it could follow the connection, the connection seemed to disappear when the device is paired, which means this experiment has rendered no result.

The next experiment was to use the nRF connect app in Android. The nRF connect app for Android can read the services, characteristics and descriptors of Bluetooth Low Energy devices. With this app the UUID's of the Amazfit Bip device can be gathered. The services that were advertised by the device could be gathered without being paired. When the device is paired, it is also possible to send sample notifications. In order to do so, a macro function is made based on the services with its characteristics.

Based on this knowledge, an experiment with the Gattacker tool was performed. The tool first scans for devices. The information from the advertisement packets is stored. This information can be found in listing 5. With this advertisement json file it is possible to gather information from the services and clone the device on the computer. This makes it possible for the phone to connect to the computer, and send the packets to the smartwatch via the computer. This is how a Man-In-The-Middle attack could be set up. However, with the Amazfit Bip this scenario was not possible. The advertisement data could be gathered with Gattacker. The next scan to gather the service information did not finish. Some research showed that this could be because the device requires BLE link-layer pairing [1]. The characteristics are secured, and scanning for services with Gattacker triggers a SMP pairing request, which is not supported by Gattacker. This also makes it impossible to perform a MITM-attack.

```
cat d04304f85f53_Amazfit-Bip-Watch.adv.
    json
{
   "id": "d04304f85f53",
   "eir": "0201061
       bff5701007c5cf5b6e5348fa244fbdd
   cd22f7bbca02d04304f85f53",
   "scanResponse": "1209416
       d617a666974204269702057617463680
   302e0fe",
   "decodedNonEditable": {
      "localName": "Amazfit Bip Watch",
      "manufacturerDataHex": "5701007
          c5cf5b6e5348fa244fbddcd22f7b
      bca02d04304f85f53",
      "manufacturerDataAscii": "W |\\ 4
          D \" C _S",
      "serviceUuids": [
         "fee0"
      ]
   }
}
```

Listing 5. Advertising data of the Amazfit Bip.

The final experiment was to read the live data from the Amazfit. The tool of Andrey Nikishaev [7] was used for the MiBand 2, which is manufactured by the same company as the Amazfit Bip. The information from the nRF connect app was needed to use the

Python script and run the experiment for the Amazfit Bip.

The UUIDs of the services on the Amazfit Bip where as follows:

- Generic Access: 0x1800
- Generic Attribute: 0x1801
- Device Information: 0x180A
- Unknown Service: 00001530-0000-3512-2118-0009af100700
- Unknown Service: 0000fee0-0000-1000-8000-00805f9b34fb
- Unknown Service: 0000fee1-0000-1000-8000-00805f9b34fb
- Heart Rate: 0x180D
- Alert Notification Service: 0x1811
- Immediate Alert: 0x1802
- Unknown Service: 00003802-0000-1000-8000-00805f9b34fb

In the Unknown Service with UUID 0000fee1-0000-1000-8000-00805f9b34fb an Unknown Characteristic with UUID 00000009-0000-3512-2118-0009af100700 and the Descriptor 0x2902 was found. These values matched with the values of the Main service UUID, Authentication Characteristic UUID and the Notification descriptor of the MiBand 2.

More interestingly, the other UUIDs of the Amazfit Bip could be matched to the UUIDs of the MiBand 2. Therefore, modifications of the Python script did not have to be made. When running the script, the information and the live sensor data from the watch could be retrieved. When the initialisation was done from the computer, the smartwatch got a pairing request, which had to be accepted.

Lastly, it was noticed that the Bluetooth MAC-address changed after a factory reset. When searching for an explanation for this, the term LE privacy came up [3]. This technique is created to protect the owner of the device from being tracked when moving. The connection between the smart device and the phone remains established by a Identity Resolution Key. Although this could be an explanation for the change of the MAC-address, it is not likely that this is the cause. When the device is reset, and the MAC-address is changed, the connection with the phone is lost and a new pairing process has to be performed.

5.2.2 Lemfo KW18

The hcitool was used to scan for the KW18. Notable is that the KW18 uses two Bluetooth MAC-addresses. One for Bluetooth and one for Bluetooth Low Energy. The Lemfo KW18 uses BLE to receive notifications from the mobile phone.

When analysing the btsnoop log file with Wireshark, the services of the device could be discovered. The device offers, in addition to the expected functions of smartwatches, services like Hands Free Profile (HFP), and Human Interface Device Profile (HID).

The nRF connect app shows the services, characteristics and descriptors of the device. It was possible to send a fake call notification to the watch. Therefore, the watch needs to have a connection with the phone that runs the nRF connect app.

The Gattacker tool used for cloning the device and performing a MITM-attack has the same result on the KW18 as well as the Amazfit Bip. The advertisement data could be logged on the computer, but retrieving service data to clone the device was not possible. The information from the advertising packet could be found in listing 6.

```
cat e0a0637fb2c2_KW18.adv.json
{
   "id": "e0a0637fb2c2",
   "eir": "02011a05094b573138",
   "scanResponse": null,
   "decodedNonEditable": {
      "localName": "KW18",
      "manufacturerDataHex": null,
      "manufacturerDataAscii": null,
      "serviceUuids": []
   }
}
```

Listing 6. Advertising data of the Lemfo KW18.

5.2.3 Lemfo LEM8

The Lemfo LEM8 was discovered by the hcitool. It was noticed immediately that the device differs from the other two, because this smartwatch did not use BLE technology. Because this limited the possibilities to use previous methods to intercept and manipulate the device, further experiments are not performed.

6 Discussion

This report explores which data is stored on the investigated smartwatches and how this data could be an information leakage risk in a business environment. This is done by investigating three different smartwatches, which are manufactured by Chinese vendors. From the questionnaire that was taken, it has become clear that most of our respondents wear smartwatches from Apple and Fitbit. The functions of these smartwatches are similar to the investigated smartwatches. This research has shown that the investigated smartwatches are not developed with security in mind, and that it is possible to gain access to the data on the device without root permissions.

During the research, data extraction has worked for only two devices: the Lemfo KW18 and the Lemfo LEM8. The Xiaomi Amaxfit Bip resets the data on the watch when connecting it to an unknown device. The data from the Lemfo devices could be extracted with third-party tools. Without these tools, extraction would be much more difficult. After extracting the data from the watch, data still exists on the device. None of the investigated smartwatches allowed for manipulation of the data.

Sending fake data with the tools that were used only worked for the Lemfo KW18, which accepted a fake call notification, and the Xiaomi Amazfit Bip, which accepted text, mail, calendar, and call notifications. It can be determined that this possible attack vector only worked on these devices because they are connected with BLE technology.

The Dutch National Cyber Security Center published a document with guidelines for mobile devices [5]. Although these guidelines are not specific for the use of smartwatches in an organisation, some of these best practices that are described in the guidelines could be used to prevent information leakage from smart devices.

7 Conclusion

The result of the questionnaire made it clear that half of the respondents wore a smartwatch, which they mostly used to read e-mail-, agenda-, text- and Whatsapp notifications. This information in the notifications could contain sensitive business information.

The specifications of the smartwatches were analysed. With the USB connection and the Bluetooth connection, the information on the devices could be gathered. On the Lemfo LEM8 and the KW18 the USB connection is investigated. The Amazfit Bip and the LEMFO KW18 both have a Bluetooth Low Energy connection, which is investigated as well in this research. The LEMFO LEM8 is equipped with Bluetooth 4.1, WiFi and a 4G LTE connection.

Both the USB connection and the Bluetooth connection were investigated. Through the USB connection, the data which was stored on the smartwatches can be extracted without the need of any additional permissions. Information is stored in plain text on the device. With the Bluetooth scenario it was possible to read live data from the Amazfit. However, it is required to pair the Amazfit with the computer, and thus go through the authentication process of the pairing process. Pairing the laptop with the Amazfit was a necessity because the watch can only be connected to one device at once. With the nRF connect app, it was possible to read service data from the Lemfo KW18 and the Amazfit Bip. Besides that, it was possible to send fake notifications through the app.

It was found that a data dump could be extracted from the Lemfo devices. The Xiaomi Amazfit resets its data when it connects to a new device. On the Lemfo devices contact information, messages, for example from WhatsApp, as well as heart rates could been extracted.

It was possible to extract the data from the Lemfo KW18 and the Lemfo LEM8. When the attempt to tamper with the data was made and when data was restored, the device got corrupted and had to be reset to its initial state. For both the Lemfo KW18 and the Amazfit Bip it was possible to send fake notifications. Besides sending fake data, live data from the Amazfip Bip could be retrieved on the computer using BLE when the device was paired. To conclude, it is possible to retrieve data from the watches that were investigated, but it was not possible to tamper with the data, which is stored on the watches. However, for the Lemfo KW18 and Amazit Bip it is also possible to send fake notifications to the device.

Some smartwatches could store information from the phone in their own memory. If this memory is read, information can be leaked. When smartwatches are paired with a company phone, the leakage risk of company data is present. Awareness of this data on the watch is important. Companies could use a business policy on smartwatches in a business environment to control the data on the watch. However, just a few smartwatches have security capabilities present.

8 Future work

As mentioned in the discussion, the smartwatches researched differ from each other and this makes it hard to come up with one generic method to gather the information on the device. This study can be repeated with different points of view.

Firstly, smartwatches can be categorised on different chipsets. Kong's study [20] described the working of MediaTek chipsets. Apple watches run on an ARM chipset, which has different characteristics. The operating system of the device is likely to have influence in these characteristics as well. It could be interesting to develop a generic method to test security or extract data from smartwatches that have the same characteristics.

Secondly, the security functions that are present on the smartwatches could be investigated. It could be interesting to see how a pin code could protect the data, or how data on the smartwatch could be encrypted. How data can be recovered after a remote wipe action is another way of tackling this topic.

Lastly, more research could be done about the different connection techniques that are used between the smartwatch and the mobile phone. In the present study a few techniques have been researched, but the Bluetooth implementation knows many variations. A more in depth investigation on this topic could be done.

References

[1] Scan by name doesn't finished. https://github.com/securing/gattacker/issues/22, note =.

[2] Ubertooth. https://github.com/greatscottgadgets/ubertooth/, note =.

[3] Understanding bluetooth security. https://duo.com/decipher/understanding-bluetooth-security, note =.

[4] Android mind reading: Memory acquisition and analysis with limeand volatility. https://digital-forensics.sans.org/summit-archives/2012/android-mind-reading-memory-acquisition-and-analysis-with-lime-and-volatility.pdf, 2012. [Online; accessed 31-12-2018].

[5] Beveiligingsrichtlijnen voor mobiele apparaten. https://www.ncsc.nl/actueel/whitepapers/beveiligingsrichtlijnen-voor-mobiele-apparaten.html, November 2012. [Online; accessed 24-1-2018].

[6] nrf connect macros (currently android only). https://devzone.nordicsemi.com/b/blog/posts/nrf-connect-macros-currently-android-only, July 2017. [Online; accessed 29-1-2018].

[7] How i hacked my xiaomi miband 2 fitness trackerŁŁa step-by-step linux guide. https://medium.com/machine-learning-world/how-i-hacked-xiaomi-miband-2-to-control-it-from-linux-a5bd2f36d3ad, May 2018. [Online; accessed 20-1-2018].

[8] Statistics & facts on wearable technology. https://www.statista.com/topics/1556/wearable-technology/, 2018. [Online; accessed 30-12-2018].

[9] The strava heat map and the end of secrets. https://www.wired.com/story/strava-heat-map-military-bases-fitness-trackers-privacy/, 2018. [Online; accessed 02-01-2019].

[10] SEC 575. https://www.sans.org/course/mobile-device-security-ethical-hacking, 2019. [Online; accessed 05-01-2019].

[11] Stephanie Hui-Wen Chuah, Philipp A Rauschnabel, Nina Krey, Bang Nguyen, Thurasamy Ramayah, and Shwetak Lade. Wearable technologies: The role of usefulness and visibility in smartwatch adoption. *Computers in Human Behavior*, 65:276–284, 2016.

[12] Jiska Classen, Daniel Wegemer, Paul Patras, Tom Spink, and Matthias Hollick. Anatomy of a vulnerable fitness tracking system: Dissecting the fitbit cloud, app, and firmware. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(1):5, 2018.

[13] Joey Dreijer and Mathijs Houtenbos. Android worn research project regarding android wear forensics. 2015.

[14] Golem. Universal readback extractor for mtk feature watchphones. https://forum.xda-developers.com/smartwatch/other-smartwatches/readback-extractor-mtk6260-firmware-t3289272. [Online; accessed 24-1-2018].

[15] Carles Gomez, Joaquim Oller, and Josep Paradells. Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. *Sensors*, 12(9):11734–11753, 2012.

[16] Shaikh Shahriar Hassan, Soumik Das Bibon, Md Shohrab Hossain, and Mohammed Atiquzzaman. Security threats in bluetooth technology. *Computers & Security*, 74:308–322, 2018.

[17] Mengfei He, Junbin Fang, Zoe L Jiang, SM Yiu, KP Chow, and Xiamu Niu. Digital forensic on mtk-based shanzhai mobile phone with nand flash. In *Proceedings of the First International Conference on Digital Forensics and Investigation, Beijing, China*, volume 2123, page 110, 2012.

[18] Sławomir Jasek. Gattacking bluetooth smart devices. *SecuRing, Tech. Rep.*, 2016.

[19] Swati Khandelwal. Researchers uncover new attacks against lte network protocol. https://thehackernews.com/2018/06/4g-lte-network-hacking.html, June 2018. [Online; accessed 24-1-2018].

[20] Joe Kong. Data extraction on mtk-based android mobile phone forensics. *Journal of Digital Forensics, Security and Law*, 10(4):3, 2015.

[21] Kevin Olivieri. Management and security implications of the smartwatch at work. https://securityintelligence.com/management-and-security-implications-of-the-smartwatch-at-work/, March 2015. [Online; accessed 31-12-2018].

[22] Reza Rawassizadeh, Blaine A Price, and Marian Petre. Wearables: Has the age of smartwatches finally arrived? *Communications of the ACM*, 58(1):45–47, 2015.

[23] Bluetooth S.I.G. Proprietary information security, bluetooth low energy.

[24] Telecompaper. Momagic to offer mediatek mre app store. https://www.telecompaper.com/news/momagic-to-offer-mediatek-mre-app-store--864423, March 2012. [Online; accessed 29-1-2018].

[25] Jeroen Van Kessel and Jan Laan. Ant+ device forensics. 2015.

9 APPENDICES

9.1 Questions from questionnaire

#	Question:	Purpose
1	Do you wear a smartwatch?	Filter out the respondents that do not have a smartwatch.
2	Which smartwatch model do you have in your possession?	To get an indication of the smartwatch models from the respondents.
3	Do you use your smartwatch for different purposes when your are not at work?	To research whether smartwatches are used for special purposes in a business environment.
4	Smartwatches help me to work more effectively	To gauge the opinion of the respondents about the usefullness of smartwatch in a business environment.
5	I personally use my smartwatch in a working environment for	To get an overview of purposes smartwatches are used.
6	Do you think there is sensitive data (company data) on your smartwatch because of the way you are using it?	To get an overview of whether smartwatch users are familiar with the risks of data leakage are when using a smartwatch in a working environment.
7	Do you have a pincode on your smartwatch?	To see whether respondents have taken precautions to protect the data on the smartwatch.
8	Have you taken any precautions in case you lose your smartwatch or when it gets stolen? (Like enabling remote control)	To see whether the respondents have taken any other precautions than a pincode to protect the data that is stored on the smartwatch.
9	What precautions did you take?	See Q8.

TABLE 3

Overview of composed questions to get an indication of smartwatch usage in a business environment.

9.2 Answers from questionnaire

9.2.1 Answers for question 1, 2, 3, 4

Do you wear a smartwatch?	**Which smartwatch model do you have in your possession?**	**Do you use your smartwatch for different purposes when your are not at work?**	**Smartwatches help me to work more effectively**
Yes	Fitbit charge 2	No	Disagree
Yes	Fossil Hybrid Q & Apple Watch Series 4	No	Neutral
No	Garmin Edge (not really a smartwatch)	No	Neutral
Yes	Google wear	No	Neutral
No	Niet van Toepassing	No	Neutral
No	None	No	Neutral
No	None	No	Disagree
No	TicWatch E	No	Neutral
No		No	
No		No	
No		No	
No		No	Neutral
No		No	
No		No	Neutral
No		No	
No		No	
No		No	Neutral
No		No	
No		No	Disagree
No		No	
No		No	Neutral
No		No	
No		No	
yes	apple	Yes	Agree
Yes	Apple	Yes	Neutral
Yes	Apple series 4	Yes	Neutral
Yes	Apple watch 2	Yes	Strongly agree
Yes	Apple Watch 3	Yes	Neutral
Yes	Apple Watch S2	Yes	Neutral
Yes	Apple Watch Serie 3	Yes	Strongly agree
Yes	Apple Watch Series 3	Yes	Disagree
Yes	Apple watch series 4	Yes	Agree
Yes	Apple Watch v2	Yes	Agree
No	Fitbit	Yes	Neutral
Yes	Fitbit Charge 2	Yes	Agree
Yes	Fitbit Charge 2	Yes	Agree
Yes	FitBit HR	Yes	Disagree
Yes	Fitbit Ionic	Yes	Agree
Yes	Fitbit Versa	Yes	Neutral
Yes	Galaxy Gear 3	Yes	Disagree
Yes	Garmin forerunner 235	Yes	Strongly disagree
Yes	mi band 3	Yes	Neutral
Yes	Pebble Steel	Yes	Strongly agree
Yes	Skagen Hybrid	Yes	Disagree
Yes	Suunton Ambit3	Yes	Neutral

9.2.2 Questions 5, 6, 7

I personally use my smartwatch in a working environment for	**Do you think there is sensitive data (company data) on your smartwatch because of the way you are using it?**	**Do you have a pincode on your smartwatch?**	**Have you taken any precautions in case you lose your smartwatch or when it gets stolen? (Like enabling remote control)**
Monitoring steps and activity	No	No	No
	No	Yes	Yes
	Yes	No	No
Reading text message / whatsapp message notifications	No	Yes	Yes
Geen van de bovenstaande.	Uncertain	No	No
Do not have one	Uncertain		
I do not use a Smartwatch	Uncertain	No	No
Reading e-mail pop-ups, Agenda notifications, Reading text message / whatsapp message notifications	Yes	No	No
	Uncertain		
	No	No	Yes
Reading e-mail pop-ups, Agenda notifications, Reading text message / whatsapp message notifications	No	Yes	No
Reading e-mail pop-ups, Agenda notifications, Reading text message / whatsapp message notifications	No	Yes	Yes
Reading e-mail pop-ups, Agenda notifications	Uncertain	Yes	No
Reading e-mail pop-ups, Agenda notifications	Yes	Yes	Yes
Reading e-mail pop-ups, Agenda notifications, Reading text message / whatsapp message notifications, Sleep tracking, Wokrout tracking, Silent Alarm clock	Uncertain	Yes	Yes
Reading e-mail pop-ups, Agenda notifications	Yes	Yes	No
Reading e-mail pop-ups, Agenda notifications, Reading text message / whatsapp message notifications	Uncertain	Yes	Yes
Reading e-mail pop-ups, Agenda notifications, Reading text message / whatsapp message notifications, To check what time it is	Yes	Yes	Yes
Agenda notifications, Reading text message / whatsapp message notifications, Reminders for standing/moving	Yes	Yes	Yes
Reading text message / whatsapp message notifications	No	Yes	Yes
	No	No	No
Agenda notifications, Reading text message / whatsapp message notifications, call notifications	Uncertain	No	No
Agenda notifications	No	No	No
Every hour it tells me to walk if i didn't. I disabled all notifications because they are rather a distraction than something that increases effectivity.	No	No	No
Reading text message / whatsapp message notifications, Receiving/Accepting Calls	No	No	No
Reading text message / whatsapp message notifications	Uncertain	Yes	No
I don't use it at work.	No	Yes	Yes
	No	No	No
Agenda notifications	Uncertain	No	No
Reading e-mail pop-ups, Agenda notifications, Reading text message / whatsapp message notifications	Yes	No	No
Reading text message / whatsapp message notifications	No	No	No
Sport data	No	No	No

9.2.3 Question 8

What precautions did you take?
None, there should be no confidential data on it.
Op afstand wipen
Will lock
Niet van toepassing
Haven’t taken any precautions yet, but I do intend to read up on any security measure there is and implement them
I dont use one.
Find my watch, remote erase function (Apple)
6-digit watch pin is likely enough
Wipe after X amount of failed logins
Reset after pin code is incorrect 10 times. Connected to my iCloud account so I can reset and lock it remotely.
Blokkeren op afstand. Instant wipe
PIN code is enforced
The apple wacht automatically locks with a 6 digit pin when taken of the wrist
location tracking, standard apple precautions
-
N/A
Find my watch and try to have the amount of PII on it as low as possible.
This question was good for security awareness... The problem is that currently the watch does not have the option to activate certain precautions. Only thing I could think of is disabling notifaction. There should be a mechanism that locks the watch when there is no connectivity with your phone or something else....

9.3 Common Android partitions

/boot	The boot partition contains a kernel image and a RAM disk combined via mkbootimg.
/system	The system partition mainly contains the Android framework.
/recovery	The recovery partition stores the recovery image, booted during the OTA process. If the device supports A/B updates, recovery can be a RAM disk contained in the boot image rather than a separate image.
/cache	The cache partition stores temporary data and is optional if a device uses A/B updates. The cache partition doesn't need to be writable from the bootloader, only erasable. The size depends on the device type and the availability of space on userdata.
/userdata	The userdata partition contains user-installed applications and data, including customization data.

TABLE 4
Typical Android partitions and their corresponding purpose from source.android.com