Wearables Security and Privacy

**Jorge Blasco, Thomas M. Chen, Harsh Kupwade Patil
and Daniel Wolff**

Abstract Wearable devices equipped with various embedded sensors are finding many applications in health care and other sectors. As a relatively new class of mobile computing, there is little experience with security and privacy problems. This chapter aims to bring attention to these important but somewhat overlooked issues. We describe the components in wearables (sensors, processors, software, and communications) and highlight the security issues related to wireless protocols, vulnerabilities, and privacy.

1 Introduction

In the past decade, smartphones have become a ubiquitous platform for mobile computing, allowing users to carry around serious computing power and always-on Internet connectivity [31]. Wearable devices extend mobile computing to be worn on the body which offers some appealing advantages: they can be carried around conveniently and continuously; they can be operated mostly hands-free; they can be highly

J. Blasco
Royal Holloway, University of London, London, UK
e-mail: jorge.blascoalis@rhul.ac.uk

T. M. Chen (✉) · D. Wolff
City, University of London, London, UK
e-mail: tom.chen.1@city.ac.uk

D. Wolff
e-mail: Daniel.Wolff.2@city.ac.uk

H. Kupwade Patil
San Jose Laboratory, LG Electronics, Santa Clara, CA, USA
e-mail: harsh.patil@lge.com

H. M. Ammari (ed.), *Mission-Oriented Sensor Networks and Systems: Art and Science*, Studies in Systems, Decision and Control 164,
https://doi.org/10.1007/978-3-319-92384-0_11

personalized in a variety of form factors; and they can incorporate an array of sensors to measure health signs [32, 45] and personal activities [16, 36, 38].

Wearables are becoming increasingly popular in sectors including infotainment, fitness, health care, and industry [15]. Statistica [63] estimates that 85 million wearables were shipped in 2015, which will increase by 58% to 135 million in 2016, and then to 190 million in 2017. Gartner [24] predicts that 50 million smartwatches, 35 million wristbands, 24 million sports watches, and 21 million other fitness monitors will be sold worldwide in 2016. The numbers do not include wearable systems specialized for military applications [71].

Wearables for infotainment include smart glasses, heads-up displays, and smartwatches. Fitness and healthcare applications involve wristbands, smart garments, chest straps, and sports watches. Wearables for industry and military applications include head-mounted displays and hand-worn terminals. Other forms of wearable devices are gloves, shoes, contact lenses, armbands, rings, caps, bracelets, and earbuds. Wearables are often designed with multiple functions, e.g., smartwatches and wristbands can monitor fitness, make contactless payments, receive or send messages, wirelessly unlock doors, and perform many more things depending on software apps.

Although wearables have certain advantages over smartphones, wearables are more likely to complement smartphones than replace them. Wearables extend computing to the body but are constrained by their often small size and mobility requirements [19]. They typically must be designed to minimize battery power usage [61]. Their hardware resources are limited usually in terms of memory and computing. Their wireless communication range is short mainly to save energy. For these reasons, they often work with smartphones to take advantage of the phone's greater computing and communications capabilities. An example of a healthcare scenario is shown in Fig. 1, but this is not a unique configuration. In this example, a smartphone may act as a hub to collect and process data from wearable sensors [42, 69]. Hubs have relatively large data storage, powerful processors, and broadband Internet connectivity. Hubs may carry out lightweight signal processing on the data and transmit

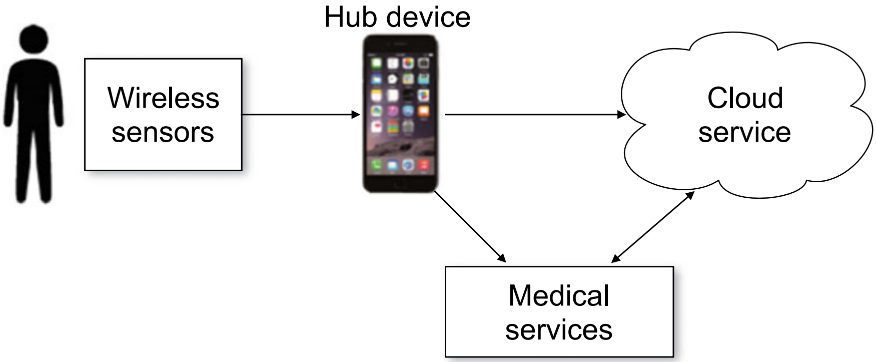

Fig. 1 A healthcare scenario

a fraction of the data to cloud servers for powerful analysis and long-term storage. Data may be shared with authorized medical services (e.g., doctors, hospitals, etc.). In the long term, wearables will be more standalone devices, as suggested by recently introduced smartwatches with Long-Term Evolution (LTE) cellular capabilities.

In the bigger picture, wearables (and smartphones) will be a part of the expanding Internet of Things (IoT) [51]. The IoT will be made up of a massive number of interconnected "smart" objects with sensing, communications, and information processing capabilities [47]. However, IoT solutions are being designed with security as a secondary consideration [53]. Security and privacy concerns for the IoT are relevant to wearables as well [67]. For example, personal health data collected by a wearable might be stolen for malicious purposes, or a vulnerability in a wearable device might be exploited by ransomware to force the owner to pay a ransom. Some wearables are used as authentication devices (e.g., for payments) which make them attractive targets for criminals.

Although research in wearables has been ongoing for decades, they have become mainstream popular with consumers only in recent years [52, 62]. There is little experience with security issues at the current time. Wearables increase the risk of certain security and privacy issues because of the following reasons:

- Wearables have a variety of biosensors, which can collect a great amount of personal data about a person [59];
- Wearables are worn constantly so a person may be monitored continuously;
- Wearables are always network-connected and accessible;
- Wearable devices are often designed for functionality and price instead of security.

In a real sense, wearables are the most intimate "personal" computing devices because they know a person's activities and physiology. It is easy to see that wearables will be attractive targets for criminals, not only for the valuable personal data stored in them but also for other possible attacks:

- Attack scenario 1: Wearables are used for access control (to open locks or log into computer accounts). A wearable is identified by a unique cryptographic key stored in memory. A criminal steals a wearable to gain entry to a victim's house or bank accounts.
- Attack scenario 2: A criminal gains access to the sensor data in a wearable to steal a victim's biometric data, e.g., facial image, voice pattern, and heart rate data. Using the stolen biometric data, the criminal carries out identity theft by masquerading as the victim.
- Attack scenario 3: A criminal eavesdrops on wireless transmissions from the wearable to steal personal data.
- Attack scenario 4: A criminal takes control over the wearable device (e.g., locks the wearable) and extorts the victim for money in return for giving back control.
- Attack scenario 5: A criminal takes control over the wearable device, perhaps with malware, and uses its resources for malicious purposes, e.g., spam, botnet, or a stepping stone to launch attacks on other devices.

The aim of this chapter is to bring more attention to security and privacy issues for wearable devices. Section 2 begins with a description of wearable devices and their components. Section 3 examines the security of common wireless protocols that are being implemented in wearables. Section 4 describes the vulnerabilities of wearable devices. Finally, Sect. 5 reports on privacy issues.

2 Wearable Devices

What is a wearable device? Wearables are a broad class of mobile computing devices with significant power and size limitations imposed by the form factors. It may be easiest to think of traditional wearable objects—such as clothes, watches, rings, glasses, and headgear—and add computing and communications capabilities to make a wearable device. Thus, in contrast, smartphones are not in the class of wearable devices because phones are traditionally thought to be "carried" but not "worn." Like any computer, wearables have processors, memory, and software. They may or may not be connected to the Internet, depending on their application. Since they are worn continuously and close to the body, they tend to include an interesting array of sensors for monitoring a range of biosignals [59]. Valuable physiological data can be collected over long time frames that can be analyzed for baseline patterns, anomalies, and gradual progression of certain symptoms.

In this section, we describe four major components in wearable devices: sensors, signal processing, processors, and software. While this section is intended mostly for background, security risks and vulnerabilities are pointed out where appropriate.

2.1 Sensors

A wide variety of sensors can be accommodated in wearable devices [10, 42]. The cost-effective production of small sensors is now possible due to technological advances in microelectronics, materials, optics, and miniaturization. Typical wearable sensors are noninvasive, i.e., work outside of the human body, and directly on the skin or in very close proximity. Invasive sensors are preferable for measurements of internal processes (e.g., bile sensors [9]) but involve surgical implantation or ingestion which are naturally unappealing.

The description of sensors here aims to be comprehensive for two reasons. First, the variety of sensors embedded in wearables is one of the major differences between wearables and traditional computers (including smartphones). Second, the data collected from sensors poses new security risks such as loss of privacy of very personal data (related to physiology, medical conditions, and daily activities) and valuable biometric data that might be stolen for purposes of identity theft.

2.1.1 Light Sensors

Cameras: Digital cameras are optical sensors for taking images or videos, combined with other sensors, special circuitry, and sophisticated signal processing for enhancing the picture quality (e.g., to compensate for low light, shaking, and motion, as well as recognize faces). They are commonplace now in smartphones, smart glasses, and other wearables. A wide range of applications include infotainment, augmented reality, and biometrics (face, retina, and fingerprint recognition).

Cameras are used in older types of fingerprint scanners to capture an image, and then the algorithms analyze the light and dark areas to recognize patterns such as ridges. An array of LEDs provides lighting for the fingerprint at scan time. This type of optical fingerprint scanner has been shown to be vulnerable to spoofing by high-quality images of stolen fingerprints. More modern fingerprint scanners are capacitive which are more difficult to fool.

Face recognition technology has been around for several decades, and many techniques are available, e.g., Viola–Jones algorithm, principal component analysis, independent component analysis, linear discriminant analysis, and so on. Face recognition is not as popular for smartphones as fingerprint recognition perhaps because face recognition is generally less reliable (affected by shadows, occlusions, and so on) and easier to spoof in the sense that faces are easier to steal than fingerprints.

For biometrics, iris patterns (the colored ring in the eyeball between the central pupil and the sclera) are appealing because they do not change after age two. While the color of the iris is determined by genetics, the patterns in the ligaments of the iris are created by random tissue folding during gestation and are unique to each eyeball. Also, there are 225 different points of comparison that are unique to each iris, compared to 40 in a fingerprint. In general, a near-infrared (NIR) light is shown into the eye because it does not cause discomfort, unlike visible light. A separate camera is used to capture the image because standard digital cameras include infrared-blocking filters. Alternatively, some iris recognition systems look at the pattern of blood vessels in the white part of the eye.

Cameras offer a noncontact approach to measuring respiratory rate, in contrast to contact approaches requiring sensors on the chest and abdomen to measure movements there. Generally, cameras capture a video of a person in visible or infrared light, and the frames are analyzed to pick out the rhythmic movements indicative of exhalation and inhalation.

As a potential point of attack, cameras are an attractive target for criminals. Gaining access to the camera can allow theft of highly personal images and biometric data.

PPG: The photoplethysmograph (PPG) measures the pulse wave as the volume change of blood [64]. It takes advantage of the fact that blood absorbs infrared light. Typically, light is emitted by one or multiple LEDs on the skin; a photodetector on the same side will detect the scattered light or a photodetector on the other side will detect the transmitted light. Each time the heart beats, a blood pressure pulse is generated and propagated in the blood vessel. A local increase of blood pressure causes an increase in light absorption and attenuation of the light transmitted through

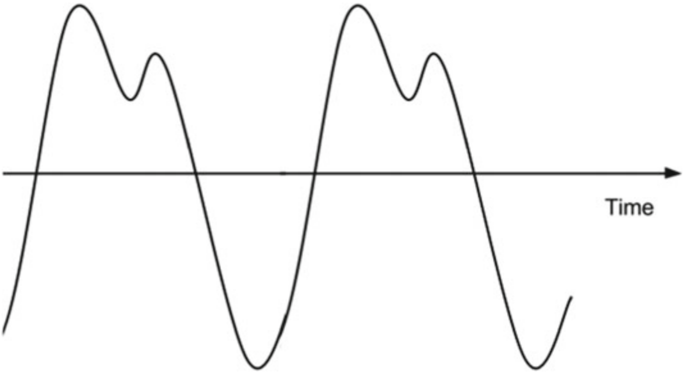

Fig. 2 An example of PPG signal

the tissue or reflected. An example of a PPG signal is shown in Fig. 2. Common operating wavelengths are between 510 mm (green) and 920 mm (infrared). Green works best on light skin and normal temperatures, whereas longer wavelengths are better for dark skins or cold temperatures. PPG is useful for monitoring heart rate [6], blood oxygen saturation (SpO_2), blood pressure, and stroke volume [54].

PPG sensor data may not be that valuable to criminals as health data, but heart rate is starting to be used for biometric authentication. PPG data may therefore be targeted for identity theft.

Pulse oximeter: A pulse oximeter is a device usually on the fingertip or earlobe (for their small capillaries) that works in a similar way as PPG. Two wavelengths of light are shown through the finger or earlobe to a photodetector on the other side to measure the fraction of oxygen saturation level in blood. The two wavelengths measure the absorption coefficients due to the difference in concentration of hemoglobin and deoxyhemoglobin levels in blood.

Blood pressure: The traditional method of measuring blood pressure is the sphygmomanometer, an inflatable cuff that squeezes the upper arm. Wearables offer a challenge to measure blood pressure with a much smaller apparatus. One approach is a cuff around the finger that applies a varying pressure. At the same time, infrared light is shown through the finger to a photodiode. Since the wavelength is primarily absorbed by hemoglobin, the light intensity fluctuations give information about the area of the finger cross section occupied by blood. The volume of the blood is related to pressure, so the light intensity can be related to arterial blood pressure.

Blood glucose: Light is one of the means to measure blood glucose concentration (other methods are described later). Diabetes has no immediate cure, and thousands of people are diagnosed each day. There are many options for monitoring glucose levels [66]. Light sensors that fit within a wearable device offer a noninvasive way that is clearly preferable to traditional invasive and painful ways.

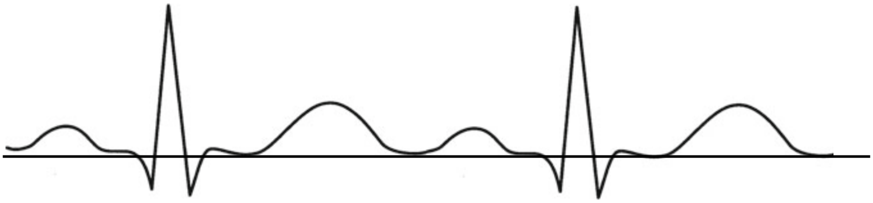

Fig. 3 An example of ECG signal

2.1.2 Electrical Sensors

ECG: Many wearable sensors focus on monitoring the cardiovascular system because of the electrical activity of the heart [59]. The most familiar electrical sensor is the electrocardiogram (ECG) consisting of two or more metal electrodes that must be in direct contact with the skin, usually facilitated by a gel for a proper connection [14]. They can be placed across the chest, wrists, and ankles. ECG electrodes measure the tiny voltage changes on the skin that arise from the pattern of depolarizing and repolarizing during each heartbeat. A healthy heart has a regular progression of depolarization starting with pacemaker cells in the sinoatrial node and eventually ending in the ventricles that create a typical ECG wave as shown in Fig. 3. Repolarization is a phase when cells return to a resting negative charge.

An ECG provides a large amount of information about the structure and function of the heart. Aside from a check of general health, it is useful for diagnosis of breathing difficulties, heart problems, fainting, seizures, and emergency situations. Wearables allow continuous ECG monitoring, which is particularly useful for people who are critically ill, undergoing general anesthesia, or have an infrequently occurring abnormal cardiac rhythm.

Unfortunately, many sources of noise can corrupt ECG signals: power line interference, electrode contact noise, motion artifacts, muscle contraction (refer to electromyogram below), and electromagnetic interference from other electronic devices. Practically, it is necessary to filter out all these noise sources.

Theft of ECG data may pose a serious privacy loss because ECG can reveal a substantial amount of information about a person's health and medical condition. Also, like PPG sensor data, heart rate measured by ECG (more accurate than PPG) may be valuable for biometric authentication. ECG data should be protected against identity theft.

Respiratory rate: Respiratory rate may be derived from an ECG because it has been observed that the respiration has a modulating effect on the ECG. The technique is called ECG-derived respiration (EDR) [41].

EMG: A surface electromyogram (EMG) is performed in a similar way as ECG with multiple electrodes on the skin to measure the electric potential generated by muscle cells when these cells are electrically or neurologically activated. A surface EMG is noninvasive but provides only limited information about muscle activity.

An intramuscular EMG gives a much more informative measurement but requires insertion of electrodes through the skin into the muscle tissue.

EEG: An electroencephalogram (EEG) measures voltage fluctuations resulting from ionic current within the neurons of the brain. Typically, multiple EEG electrodes are placed in a head-worn apparatus to make contact with the scalp. Noninvasive EEG is used to diagnose epilepsy, sleep disorders, coma, stroke, encephalopathies, and brain disorders in general. However, a clinical EEG can take 20–30 min; EEG is not good at measuring neural activity below the upper layers of the brain (the cortex), and generally the signal-to-noise ratio is poor.

Like ECG data, theft of EEG data may pose a serious loss of privacy. Unlike ECG data, the EEG is not currently used for biometric authentication, so the reason for theft of EEG data is not likely to be identity theft.

GSR: Another electrical sensor is the galvanic skin response (GSR or skin conductance) sensor used to measure the electrical conductance of the skin [44]. Two electrodes are placed on the skin close to each other and pass an imperceptible current between them. The measured electrical resistance of the skin depends on the moisture or sweat produced by the skin. Sweating is controlled by the sympathetic nervous system, and GSR is sometimes interpreted as an indicator of arousal or stress.

Temperature: Finally, electrical sensors are common for measuring temperature (among other methods such as infrared detection). Electrical temperature sensors can be built using a thermistor or thermocouple. A thermistor changes resistance with temperature; the resistance is measured by a bridge circuit containing the thermistor. A thermocouple takes advantage of the property that a small voltage is generated at a junction of different conductors that is proportional to their temperature difference.

2.1.3 Electrochemical Sensors

Sweat rate: A real-time sweat rate sensor was constructed from two capacitive humidity sensors at different distances from the skin [57]. A capacitive humidity sensor consists of a nonconductive foil which is covered with gold on both sides. The dielectric constant of the foil changes as a function of the relative humidity of the ambient atmosphere, which is measured as the capacitance value. The difference between the measurements at the two humidity sensors gives an indication of water vapor flow from the skin's surface.

Sweat: As mentioned earlier, sweat contains an abundance of interesting electrolytes and metabolites. Up to now, noninvasive biosensors have been able to monitor a single analyte at a time or lack on-site signal processing circuitry. Gao et al. [23] have built a wearable containing an array of electrochemical sensors for in situ sweat analysis including glucose, lactate, sodium, and potassium ions. The glucose and lactate sensors are electrodes coated with a specific enzyme, namely, glucose oxidase, and lactate oxidase, respectively. These enzymatic sensors generate electric current proportional to the abundance of the corresponding metabolites between the working electrode and a reference electrode.

Glucose: A noninvasive method to measure the glucose level in blood would be valuable for managing diabetes [66]. Optical methods to measure blood glucose were mentioned earlier. A correlation has been found between sweat glucose and blood glucose, so some researchers have focused on sweat glucose [43]. Sweat glucose may be measured noninvasively (as described above) but measurements can be easily confounded by other factors in sweat.

2.1.4 Motion Sensors

GPS: In wearables, motion sensors can be built based on location sensors or force-based sensors. GPS is a well-known satellite system for triangulating location on Earth using signals from four line-of-sight GPS satellites [13]. GPS receivers provide a location within a few meters or so, depending on the type of GPS receiver. Exposure of location information is sometimes seen as a threat to privacy.

Magnetometers: Magnetometers or compasses measure the direction of the Earth's magnetic field to determine the bearing or direction of an object. Digital magnetometers are small and inexpensive, and thus suitable for embedding in almost any electronic device including wearables. A digital magnetometer is a type of force-based motion sensor that is generally embedded within other force-based sensors such as accelerometers and gyroscopes.

Accelerometers: Accelerometers are widely used in smartphones and other mobile devices to detect device orientation and serve as input to motion-based games. Commonly used accelerometers measure g-force (1 g is 9.81 m/s^2) in the three axes: x, y, and z. Four kinds of accelerometers are available: piezoelectric, piezoresistive, capacitive, and servo-type sensors. They work on the principle of generation of electricity, change in resistance, change in capacitive effect, and change in heat induction, respectively.

Accelerometers along with gyroscopes may be used to infer a person's activities. Hence, the data may be considered to be worth protecting as personal data.

Gyroscopes: Gyroscopes measure attitude and rotation. Attitude is the orientation of the gyroscope relative to a point in space. By measuring changes in attitude, gyroscopes can also measure its rotation rate.

Pedometers: A pedometer counts the number of steps walked by detecting when a body tilts from side to side, e.g., by movement of the hips, and multiplies the number of steps by the length of each step to determine a total distance traveled. Inside a pedometer, a metal pendulum swings when the body tilts to one side to make electrical contact with an electronic counting circuit, incrementing the count by one. When the body tilts back, the pendulum swings back and breaks the circuit. Other pedometers are entirely electronic, using two or three accelerometers. These are arranged at right angles that detect minute changes in force when legs move during a step.

Shoe sensors: Shoes can be fitted with pressure sensors and accelerometers to track steps or analyze gait. Pressure sensors are usually made of several thin layers of a piezoresistive material, such as silicon, that becomes more resistant to an electric current when force is put on it. The surface is connected to a Wheatstone bridge, which is designed to detect small differences in resistance.

2.1.5 Sound Sensors

Microphones: Microphones change sound waves into an electrical signal. They are inexpensive and small, so they are commonplace in many types of electronic devices. Microphones are useful for a variety of applications including voice recognition, respiration rate analysis, and emotion detection. Microphones have the drawback of capturing ambient noise as well as the interesting sound. Multiple microphones and signal processing techniques are typically used to reduce the effects of noise [11].

Microphones are often a target for criminals because access to sound may allow criminals to hear personal data or steal voice patterns for biometrics. Thus, the threats are privacy loss and identity theft.

Ultrasound: Ultrasound at frequencies above human hearing has many useful applications, e.g., fingerprint scanning. A fingerprint can be scanned by transmitting an ultrasonic pulse against the finger placed on a scanner. While some of the pulses are absorbed, the rest is bounced back to the sensor, depending on the ridges, pores, and other microstructures that are unique to each fingerprint. The sensor calculates the intensity of the returning ultrasonic pulse at different points on the scanner.

Fingerprint data should obviously be protected against theft by criminals who could use the data for identity theft.

2.2 *Signal Processing*

A wearable device will often send its sensor data to a hub device for long-term storage and heavy processing (e.g., data mining or classification). This saves memory storage and reduces energy consumption in the wearable. However, there are a number of functions that need to be carried out in the wearable, namely, signal conditioning and signal processing as shown in Fig. 4.

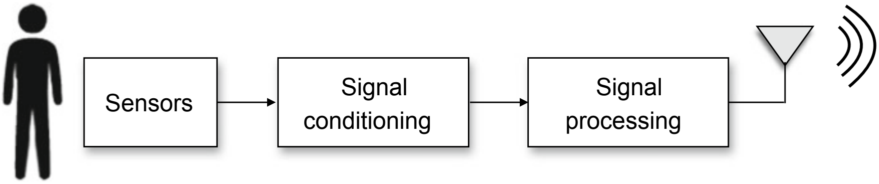

Fig. 4 Signal conditioning and signal processing in a wearable

Decisions about which functions to design into the wearable (as opposed to leave to the hub device or cloud service receiving the data from the wearable) are complicated by the following considerations:

- The total power consumed depends on the energy used by the sensors, sampling, signal preprocessing, and wireless transmission [59]. There are trade-offs, e.g., it might be more efficient sometimes to process data in the wearable instead of transmitting the data.
- Power consumption should be minimized but balanced against performance and cost, which depend on the application, for example, some applications may require a minimum sampling rate.
- Certain time-critical functions may be better to perform in the wearable, e.g., detection of imminent hazards. Another reason to carry out functions in the wearable might be unreliability of the wireless channel or cloud service.
- Wearables have a wide range of processing capabilities (from simple fitness bands to sophisticated smartwatches). Some functions may not be feasible to support in basic wearables.

Signal conditioning may include noise filtering or cancellation, signal amplification, anti-aliasing (e.g., low-pass filtering), and analog-to-digital conversion consisting of sampling and quantization [59]. Noise can be a substantial factor due to user movements, environmental noise, and changes in sensor locations (e.g., a smartwatch slipping around the wrist). Sampling frequency is another important consideration because a higher sampling rate not only improves data resolution but also increases the amount of data (and hence power consumption).

Signal processing may include data compression and lightweight classification but it is dependent on the application. Data compression reduces the total amount of data for transmission and storage, but lossy data compression achieves higher compression at the cost of discarding information that will be unrecoverable later. The compression algorithm depends on the application and what type of information should be discarded preferentially. In most cases, data should be transmitted to a hub device or cloud service for data mining and classification, but certain applications may necessitate lightweight classification to be performed in the wearable. For example, there may be applications that are sensitive to the communication delay or unreliability. In that case, algorithms for feature extraction and classification must be designed to be as efficient as possible [59].

2.3 Processors

Wearables take many forms depending on where they are worn on the body, but they are all constrained by power, memory, and computing resources. Understandably, people do not want to wear bulky, heavy equipment. At the same time, they have realistic expectations that wearables have limited functions. Hence, the microprocessors found in wearable devices have lower specifications than desktop computers and laptops, and even some smartphones.

The most common processors are based on reduced instruction set computer (RISC) in contrast to traditional complex instruction set computer (CISC) processors designed for desktop computers, exemplified by Intel's x86 platform. The RISC approach chooses a set of simpler instructions than CISC in order to reduce the number of processor cycles required to perform each instruction, resulting in smaller hardware and lower power consumption. Some high-end wearable devices have a separate coprocessor to off-load the processing of sensor data from the main processor. A coprocessor referred to sometimes as a "sensor hub" is useful when the device has a great amount of sensor data that needs to be analyzed together in real time, requiring constant CPU attention.

While there have been many RISC processors (e.g., MIPS, SPARC, and PowerPC), processors based on the ARM architecture licensed from ARM Holdings have become the most popular, adopted in wearables as well as iOS and Android smartphones and tablets [39]. The ARM Cortex-M family is well suited for low-end wearables due to its small form factor and low power requirements. For instance, the Cortex-M3 processor is used in the Pebble watch, Fitbit fitness bands, and Arduino Flora. The Cortex-M4, Cortex-M7, and Cortex-M33 processors integrate digital signal processing (DSP) and floating point operations, which are advantageous for applications such as sensor fusion and power management.

The ARM Cortex-A processor family tends to focus graphics and CPU power, compared to the Cortex-M. This tends to be found in high-end wearables such as smartwatches supporting an operating system capable of running a variety of apps and communicating with other devices (like wireless earphones or smartphones).

Many wearables use custom systems on chip (SoCs) that usually integrate multiple cores, graphics processing unit (GPU), DSP, GPS, wireless communications, and support for audiovisual sensors. Well-known examples of SoCs include the following:

- Apple's 64-bit A9 (based on ARMv8) with an integrated M9 motion coprocessor that was first implemented in the iPhone 6S and 6S Plus;
- Samsung's multi-core Exynos 9 (also based on ARMv8) appearing in the Galaxy S8 and S8+ smartphones [58];
- Qualcomm's Snapdragon Wear 2100 based on ARM Cortex-A7 [49].

The ARM Cortex-M23, Cortex-M33, and Cortex-A series processors support TrustZone technology, which dedicates a secure area on the chip called trusted execution environment (TEE) [40]. The TEE is an area for trusted resources—software, data, and hardware—separated by hardware from untrusted resources. The trusted environment can also include memory, peripherals, interrupts, and bus transactions. Common uses include the protection of authentication mechanisms, cryptography, trusted software (e.g., secure boot and electronic wallet), and biometric data. Untrusted software cannot access secure resources directly. Thus, secure resources are protected from software attacks and common hardware attacks. Context switching between secure and nonsecure environments is done in software via a secure monitor call in Cortex-A, or hardware in Cortex-M.

Apple's A7 and later SoCs are based on ARMv8 and contain a secure coprocessor called "secure enclave" that is likely using ARM's TrustZone technology. The secure enclave is known to protect data from the Touch ID fingerprint sensor. Reportedly, it has its own secure boot process to ensure security. It has a unique, unalterable ID useful for creating a temporary encryption key to encipher its memory. It also contains an anti-replay counter.

2.4 Software

Wearable devices are highly fragmented from a software perspective, with many operating systems (OSs) sharing the market [3, 33]. An OS for wearables is different from a traditional desktop OS in a number of ways: it should optimize process scheduling and power consumption; it should support the wearable's user interface; it should optimize graphics processing; and it should support the wearable's sensors input/output. Some wearables with limited functions (fitness trackers and smartwatches) do not have an operating system, whereas others need an operating system capable of supporting an ecosystem of apps.

2.4.1 Open-Source Operating Systems

Android Wear: Somewhat confusingly, Android Wear is a wearable OS that is different from the popular Android OS for smartphones [4]. Android Wear was derived from Android to be suitable for smartwatches and is mainly designed to pair smartwatches to work with Android smartphones (although version 2.0 enables Android Wear smartwatches to run native apps without the need for a smartphone nearby). Android Wear is mostly open source but Google adds a proprietary layer of services such as Google Now (for voice recognition).

Android: Android itself is not designed for wearable devices but can be modified for a wearable. Like Android Wear, Android can run on the ARM Cortex-A processor and potentially any processor supporting the Linux kernel (which Android is based on).

Tizen: Tizen is an open-source OS, also based on Linux, started by a group of companies in 2011 as an alternative to Android [65]. While it has not found success in smartphones, it has been adopted for a significant number of smartwatches. Tizen is most commonly found in Samsung smartwatches with ARM Cortex-M processors. Tizen supports apps in the Tizen app store (native apps are written in C, whereas Android apps are written in Java).

Embedded Linux: Wearables may choose embedded Linux because Linux is open source and supported on a wide variety of processors including ARM Cortex-M, Cortex-A, MIPS, and x86. Linux is a general-purpose OS, which means that apps can be developed easily, but Linux might be an overkill for a wearable designed for limited functions.

mbed OS: The open-source mbed OS is based on a real-time operating system, CMSIS-RTOS RTX [8]. Supported by ARM, mbed OS runs on a range of Cortex-M processors. For security, a supervisory kernel called uVisor helps to isolate security domains used to restrict access to memory and peripherals.

2.4.2 Proprietary Operating Systems

watchOS: Apple's watchOS is a version of its proprietary iOS customized for its Apple Watch. The original Apple Watch used a custom S1 system in package (SiP) that integrated an application processor, memory, storage, and support processors for wireless communications, sensors, and I/O controllers in a sealed package. The Apple Watch series 2 uses the S2 SiP. It is known that iOS, and hence watchOS, is based on the XNU kernel, a hybrid between BSD and Mach kernels.

Windows 10: Recently, Microsoft designed the latest Windows 10 to work across the broadest range of machines including wearables. One of its central features is the so-called "universal app" platform which means that developers can create apps that will run across all Windows devices of any size and form factor.

Pebble OS: Pebble OS was an operating system developed for the Pebble smartwatch until Pebble Technology was shut down in December 2016. Pebble OS was based on the FreeRTOS kernel, a real-time OS for embedded devices. Most of Pebble's intellectual property and staff, except hardware, were purchased by Fitbit.

LinkIt OS: MediaTek offers a proprietary LinkIt OS specialized for the Aster SoC which features low power and low cost [37]. It has a low-power standby mode, which enables always-on wearable devices to have small energy footprints. The battery life of devices can reportedly last a few days with normal usage.

WebOS: WebOS based on the Linux kernel was originally created by Palm as the successor to Palm OS. Palm was acquired by HP which released the operating system as open source under the name Open webOS. HP licensed webOS to LG Electronics in 2013 for its web-enabled smart TVs. WebOS made it into the LG Watch Urbane LTE but the current line of LG Watch Urbane (with Qualcomm Snapdragon processors) supports only Android.

WearableOS: WearableOS is a special package of the Unison RTOS (real-time operating system) [55]. A real-time OS has a deterministic preemptive kernel and a small memory footprint. WearableOS is specifically designed to minimize power consumption and support a range of sensor and wireless technologies.

3 Wireless Communications Security

Wearables take advantage of a number of wireless technologies to communicate with other devices or a cloud service [25, 32, 45]. As mentioned in Sect. 2.2, wearables have limited processing, memory, and power resources. Wearables can save resources by sending its sensor data (after preprocessing) to a device with more resources, e.g., a smartphone.

This section gives an overview of the common wireless technologies which differ in several ways: range; data rates; spectrum; error control; robustness against interference, atmospheric attenuation, and various sources of noise; and protection against eavesdropping. The IEEE 802.15 working group has developed a few standards for wireless communications applicable to wearable devices, namely, IEEE 802.15.1 (Bluetooth), IEEE 802.15.4 (Zigbee), and IEEE 802.15.6 (body area networks). Other protocols including ANT+, UWB (Ultra-wideband), NFC (near field communication), IEEE 802.11 (Wi-Fi), GPRS (General Packet Radio Service), and UMTS (Universal Mobile Telecommunications System) are also used among wearable devices. There is not much to say about IEEE 802.11, GPRS, or UMTS because they are general-purpose wireless services not designed particularly for wearables.

Wireless communications are expected to be an avenue for attackers. Wireless communications face the same security risks as wired communications (e.g., eavesdropping, data modification, packet injection, masquerade, and replay) except that attacks are easier to accomplish in the radio environment. For instance, eavesdropping on radio signals is easy for any receiver within range, whereas a wired link requires a physical tap. An unsecured wireless link may expose personal data, or worse, allow an adversary to bypass other security mechanisms, and compromise a wearable device.

As security risks are well known, each wireless technology includes security mechanisms. Cryptography is the foundation for secure communications. The standard encryption algorithm advanced encryption standard (AES) is typically employed to ensure confidentiality. However, protocols may differ in the choice of key length, block cipher mode, method for key agreement (key distribution), and calculation of MAC (message authentication code) for data integrity. Another important difference may be how devices are authenticated to each other.

For wearables, it must be kept in mind that they have very limited computation and power resources. Consequently, traditional cryptographic approaches for encryption and key establishment may not be well suited [67]. For instance, public key cryptography is considered to be too computationally demanding for wearables, and hence private key cryptography is assumed. However, this raises the question of how symmetric keys will be distributed securely.

3.1 Bluetooth

Bluetooth is standardized as IEEE 802.15.1, but the commercial technology is managed by the Bluetooth Special Interest Group (SIG) consisting of more than 30,000 companies [60]. Bluetooth is popular due to its design oriented at simple and low-cost implementation. It is widely implemented in smartphones, fitness trackers, wireless earphones, and other accessories. Bluetooth 4.0 provides a specific stack for low-power communications called Bluetooth Low Energy (BLE), also marketed as Bluetooth Smart, that is particularly relevant for wearables.

BLE utilizes 40 radio channels with 2 MHz spacing in the 2.4 GHz unlicensed band [27]. BLE communication is divided into two phases: advertising and data communication. Advertising messages use 3 out of the 40 available RF channels and allow device discovery and connection establishment. Once the advertising device (e.g., wearable) receives a connection request from the master device (such as a smartphone), the data transfer phase starts. Both paired devices can start exchanging data frames through the remaining 37 RF channels using adaptive frequency hopping. Communications between paired devices are limited between 10 m and 1 Mbps.

BLE allows one device serving as the master connected with an unlimited number of slaves to form an ad hoc piconet. A slave in one piconet can act as the master for another piconet simultaneously, thus creating a chain of networks called a scatternet.

Due to its wide adoption, Bluetooth security has been studied extensively [12, 21]. Security features include stealth, frequency hopping, authentication, and encryption.

Stealth: Devices can hide and refuse connections through non-discoverable and non-connectable modes. Normally in discoverable mode, devices reply to inquiries, letting other nearby devices discover their existence, but in non-discoverable mode, devices do not announce their presence by ignoring inquiry scans. In connectible mode, devices listen for requests to their Bluetooth address whereas in non-connectible mode, they do not allow other devices to initiate connections.

Frequency hopping: BLE uses frequency hopping spread spectrum (FHSS) to mitigate interference between devices but it helps to protect against eavesdropping. A device follows a pseudorandom sequence to hop among 37 different radio channels that are established during connection establishment [29]. In order to eavesdrop, an adversary has to determine the hopping sequence. Unfortunately, the limitations of BLE connections allow an attacker to easily get the sequence [56].

Authentication: Bluetooth has four security modes for authentication and encryption. The first three (modes 1 to 3) apply to legacy versions, while mode 4 applies to current versions. Security mode 1 is insecure with no authentication or encryption. Mode 2 (service-level enforced security) uses authentication and encryption at the service level, after a channel has been established. Mode 3 (link-level enforced security) uses authentication and encryption at the link-level connection is established. Mode 4 offers secure simple pairing (SSP) to create service-level security, similar to security mode 2.

SSP simplifies the pairing process compared to legacy Bluetooth which uses a personal identification number (PIN) to authenticate devices (not users). In comparison, SSP offers four association models that are flexible in terms of device input/output capability:

- Numeric comparison for a pair of Bluetooth devices capable of displaying a six-digit number and asking the user to enter a yes/no response on each device if the numbers match.
- Passkey entry for one Bluetooth device with input capability (e.g., keyboard) and another device with a display but no input capability.

- Just works where at least one device does not have a display or a keyboard for entering digits (e.g., headset).
- Out-of-band (OOB) for a pair of devices that support a common additional wireless or wired communication channel for device discovery and cryptographic value exchange.

SSP also improves security through the addition of elliptic-curve Diffie–Hellman (ECDH) key agreement to generate a secret symmetric key called long-term key (LTK). ECDH is a variation of the well-known Diffie–Hellman protocol [20] that makes use of elliptic-curve cryptography [35]. The Diffie–Hellman protocol allows two devices to establish a shared secret (in this case, the LTK) by exchanging public numbers over an insecure communication channel. ECDH is believed to be strong against passive eavesdropping and man-in-the-middle (MITM) attacks during pairing.

Each device generates its own ECDH public–private key pair using P-256 or P-192 elliptic curves. Each device sends the public key to the other device according to the Diffie–Hellman protocol. The devices then perform stage 1 authentication which is dependent on the association model (described above).

Bluetooth 4.2 added the secure connections feature which upgraded low-energy pairing to utilize advanced encryption standard—cipher-based message authentication code (AES-CMAC) and P-256 elliptic curve. This means that the LTK is generated based on an AES-CMAC-128 function. Also, when both BLE devices support secure connections, P-256 elliptic curves are used; otherwise, P-192 curves are used during ECDH.

Bluetooth 4.2 renamed low-energy pairing to low-energy legacy pairing. As legacy pairing does not use ECDH, it provides no eavesdropping protection and is considered broken for all pairing methods except OOB.

Encryption: BLE uses advanced encryption standard—counter with cipher block chaining message authentication code (AES-CCM) encryption [68]. AES-128 is a U.S. standard block cipher with 128-bit keys. CCM combines cipher block chaining mode with MAC authentication. The CCM mode generates an encrypted keystream that is applied to input data using the XOR operation and creates a 4-byte MAC in one operation. It is difficult for an eavesdropper to decrypt packets without intercepting packets in the initial key exchange phase.

During pairing, the LTK is generated and stored locally in each device. There is no exchange of the LTK, and therefore, pairing is not vulnerable to interception of the LTK by an eavesdropper. The link is encrypted by AES-CCM using an encryption key derived from the LTK. AES-CCM is used to provide confidentiality as well as per-packet authentication and integrity.

There is no authentication challenge/response step to verify that both devices have the same LTK or CSRK. The LTK is used to generate the link encryption key, and therefore, successful encryption implicitly provides authentication.

Bluetooth 4.0 introduced two features: low-energy private device addresses and data signing. These two features involve the generation of two keys: the identity resolving key (IRK) and connection signature resolving key (CSRK).

If BLE's privacy feature is enabled, the IRK maps a resolvable private address (RPA) to an identity address. The identity address is a static random address or a public address. The IRK allows a trusted device to determine the identity address of another device from an RPA which can be dynamic. Previously, a device would have to be assigned a static public address, and the public address could be learned during discovery. If that device remained discoverable, its location could be tracked by an adversary.

The CSRK is used to verify cryptographically signed attribute protocol (ATT) data frames from a Bluetooth device over unencrypted links. This allows a Bluetooth connection to use data signing (providing integrity and authentication) instead of data encryption (AES-CCM provides confidentiality, integrity, and authentication).

A number of vulnerabilities and attacks specific to Bluetooth are known [21]. These include the following:

- Bluebugging exploits a security flaw in the firmware of some older Bluetooth devices to gain access to the device and its commands.
- Bluesnarfing exploits a firmware flaw in older Bluetooth devices to gain access to the device.
- Bluejacking is an attack similar to phishing that consists of an unsolicited message to convince the user to respond in a certain way or add a new contact to the address book.
- Bluetooth fuzzing consists of malformed data sent to a device's Bluetooth radio and observing how the device reacts.
- Legacy pairing is susceptible to eavesdropping.
- A number of techniques can force a remote device to use Just Works SSP and then exploit its lack of man-in-the-middle protection.

3.2 Zigbee

Based on the IEEE 802.15.4 standard, Zigbee is designed for low-power wireless personal area networks (WPANs). It is intended to offer a simpler and less expensive alternative to Bluetooth or Wi-Fi for applications that do not require a high data rate (i.e., up to 250 kbps). It operates in 16 channels, each 2 MHz bandwidth, that are 5 MHz apart in the 2.4 GHz unlicensed band. It can also use regional unlicensed bands: 784 MHz in China, 868 MHz in Europe, and 915 MHz in the USA and Australia.

Commercialization is overseen by the Zigbee Alliance [2], which publishes application profiles to support interoperability between different products. Also, the alliance certifies Zigbee devices that meet power, bandwidth, and battery requirements. For instance, Zigbee devices should have a minimum battery life of 2 years and output radio power of 0–20 dBm (1–100 mW). For its low power and low data rates, the main applications of Zigbee include wireless sensor networks, embedded sensing, medical data collection, smoke and intruder warning, and building automation. However, it has not been popular for wearables so far.

Zigbee is flexible in terms of supporting star, tree, and mesh network topologies. In each topology, one node acts as a coordinator, including creation of the network. The central node in a star network must be the coordinator. The tree and mesh topologies are useful for transmitting data long distances by multi-hopping through devices acting as Zigbee routers.

The Zigbee RF4CE specification defines a low-cost communications standard that is able to provide reliable levels of connectivity for consumer electronics. It was specifically designed for applications requiring simple device-to-device control communications that do not need the full-featured mesh networking capabilities offered by Zigbee. RF4CE reduces memory size requirements and the cost of implementation. Examples of applications anticipated by the Zigbee Alliance include lighting, fan control, garage door openers, and keyless entry systems. Its purported advantages include channel agility using three channels instead of 16, a power management mechanism for all device classes, a discovery mechanism for nodes, multiple star topology, inter-PAN communication, and a security key generation mechanism.

Building on the basic security framework defined in IEEE 802.15.4, Zigbee implements most security procedures at the network and application layers, which cover key establishment, key transport, frame protection, and device management. Security is based on the AES-128 encryption cipher. Several suites combining AES-128 and MACs of various lengths are offered with increasing security levels as follows:

- no security;
- confidentiality only: AES-CTR (AES-128 in counter mode);
- authentication only: AES-CBC-MAC (AES-128 cipher block chaining message authentication code) with 32-, 64-, or 128-bit MAC;
- confidentiality and authentication: AES-CCM (same as BLE described above) with 32-, 64-, or 128-bit MAC.

A 128-bit key can be associated with either a network or a link. An initial master key must be obtained through a secure medium (transport or preinstallation). The security of the entire network depends on the master key. Link keys are derived from the master key. Link and master keys are only visible to the application layer. Various services use different one-way variations of the link key to avoid security risks.

Zigbee authentication is performed using ECMQV (elliptical curve Menzies–Qu–Vanstone), a key agreement protocol based on Diffie–Hellman using elliptic curves. It is believed to be a secure form of authentication.

One special device that is trusted by the other devices is recognized as the trust center. The trust center keeps the network key and provides point-to-point security. Ideally, devices will have the trust center address and initial master key preloaded. The trust center provides a network key to typical applications that do not have special security needs.

Many attacks on Zigbee have been investigated. Physical attacks include malicious signal interference; Zigbee can change frequency channels in the presence of interference, but it is relatively slow (Zigbee does not use frequency hopping). Physical access to a Zigbee device's RAM may access the encryption key which is often flashed on all the devices in a Zigbee network. An adversary may be able to use a special serial interface on a Zigbee device to capture the encryption keys as those keys are moved from flash to RAM during power up.

Encryption keys might be captured remotely. Zigbee radios use pre-shared keys or over-the-air (OTA) key delivery. OTA delivery may be attacked by a malicious node mimicking a node on the Zigbee network to capture packets, which can then be analyzed and decrypted using free and open-source equipment.

Replay and/or injection attacks may be able to trick Zigbee devices into performing unauthorized actions. Zigbee devices are susceptible to these types of attacks because of the lightweight design of the protocol, which has very minimal replay protection and session checking.

3.3 IEEE 802.15.6

The IEEE 802.15.6 standard specifies communications for a type of WPAN called wireless body area network (WBAN) to interconnect low-power devices that are implanted within the body or mounted on the body. WBAN is limited to a short range within the immediate proximity of a human body. A WBAN might utilize a WPAN device as a gateway to the Internet.

In order to support a variety of medical, consumer, and entertainment applications, the standard includes three physical layers: narrowband, UWB (ultra-wideband), and HBC (human body communication) in frequency bands around 400 MHz, 800 MHz, 900 MHz, and 2.4 GHz.

Three levels of security are prescribed in IEEE 802.15.6 [67]:

- level 0 unsecured communications: data frames have no encryption, data authentication, or integrity assurance;
- level 1 authentication only: frames use authentication but not encryption;
- level 2 authentication and encryption: data frames use authentication and encryption.

One of the security levels is selected during the association process where a node and a hub identify themselves to each other. A master key (MK) is established between them for unicast secured communication or a pre-shared key is activated. A pairwise temporal key (PTK) is created for each new session. For multicast secured communication, a group temporal key (GTK) is shared with the corresponding group using the unicast method.

A 256-bit key establishment is based on the Diffie–Hellman protocol with elliptic curves. The cipher-based message authentication code (CMAC) is used to derive the MK and key message authentication codes (KMAC). Initially, the node and hub have a pre-shared MK. The node initiates the association process by sending a security association frame request. The hub responds by joining, and the pre-shared MK is activated and shared between the node and hub by mutual agreement. Then, a new PTK is generated and shared.

Data frames can be transmitted in secured or unsecured communication modes. Nodes that do not require security receive all frames including beacons without validating the security information. The secured frames are authenticated and encrypted or decrypted using the AES-128 CCM mode (as in Zigbee and BLE).

3.4 ANT+

ANT is a proprietary ultralow-power protocol for wireless sensor networks from ANT Wireless, owned by Garmin [5]. It is similar to BLE but oriented toward applications with sensors. Communication range is limited to 20 m, and data rate is low (bursts up to 60 kbit/s) in the 2.4 GHz band. ANT can be used for body area networks, personal area networks, and local area networks.

ANT+ is an interoperability function added to the base ANT protocol to allow nearby ANT+ devices to work together to collect sensor data. ANT+ uses "device profiles" that specify how data is transmitted between devices, including the data format, channel parameters, and other communication parameters. For example, ANT+ enabled fitness monitoring devices such as heart rate monitors, pedometers, speed monitors, and weight scales can all work together to assemble and track performance metrics. Device profiles are shared among all ANT+ adopters, enabling any ANT+ adopter to create an interoperable device.

As a proprietary WSN protocol, not much is known about ANT+ security except that it is based on keys. ANT+ network keys are required to access the ANT+ network. Network keys are generated and provided by the ANT Alliance. Only devices with the same profiles and network keys can communicate with each other. Network keys must be requested from the ANT+ Alliance, an open special interest group of companies, after subscribing to be an ANT+ adopter.

3.5 UWB

Similar to spread spectrum, UWB spreads data across a very wide spectrum, in this case, defined to be at least 500 MHz of spectrum or 20% or more of the center frequency. As a result, the power spectral density is very low which limits the interference with conventional radio systems using the same spectrum. In the U.S., the federal communications commission (FCC) approved UWB in the 3.1–10.6 GHz range at a power level of −41.3 dBm/MHz or 75 nW. The spectrum above 3 GHz avoids overlap with GPS, cellular, and many other services.

UWB was appealing for short-range, high data rate applications but suffered a couple of setbacks. First, the IEEE 802.15.3a task group attempted to bridge competing UWB proposals from the UWB Forum and the WiMedia Alliance. The IEEE 802.15.3a task group was deadlocked for several years and eventually disbanded in 2006. Most vendors went with the WiMedia Alliance specifications using orthogonal frequency division multiplexing (OFDM). The specification divides the allowed spectrum into 528 MHz sub-bands of OFDM channels. Data rates can reach 480 Mbps at a range up to 10 m.

The second problem was competition from other high-speed wireless technologies being standardized by the IEEE 802.11 working group. In 2009, IEEE 802.11n offered a maximum single-channel data rate exceeding 100 Mbps and a theoretical maximum overall data rate of 600 Mbps using 40-MHz bandwidth with four spatial streams. Then IEEE 802.11ac, an extension of 802.11n, offered a single-link minimum of 500 Mbps and overall 1 Gbps in the 5 GHz band. IEEE and the wireless gigabit alliance (WiGig) jointly developed IEEE 802.11ad offering short-range theoretical speeds up to 7 Gbps in the 60 GHz unlicensed band. However, 802.11ad requires substantial power and is limited to line of sight. UWB also has advantages in greater resistance to noise, superior security, high jamming resistance, greater multipath immunity, low-power consumption, and high-penetration ability.

As a physical layer technology, most security issues handled in higher protocol layers are not relevant to UWB. The main security threat is eavesdropping. Because of the low average transmission power, UWB has an inherent immunity to detection and eavesdropping. An eavesdropper has to be very close to the transmitter (about 1 m) to be able to detect transmissions. In addition, UWB pulses are time modulated with codes unique to each transmitter/receiver pair. The time modulation of extremely narrow pulses adds more security to UWB transmission, because detecting picosecond pulses without knowing when they will arrive is nearly impossible.

Naturally, data will be encrypted but there is a question of whether standard encryption algorithms such as AES may consume too much power. It has been proposed to save power by pushing part of the cryptography into the physical layer by hiding the signal in the time domain [34]. The transmitter and receiver share a secret key. The key is used to randomly offset UWB pulses such that an eavesdropper cannot detect the signal coherently without knowing the key.

3.6 NFC

NFC is for short-range wireless communications (limited to 10 cm) commonly used for contactless payments. It is also used for sharing photos and files between devices, and enabling devices to act as identity authentication, e.g., keycards. Two NFC devices within 10 cm use electromagnetic induction between antennas to exchange data up to 424 kbps in the 13.56 MHz unlicensed band. As a fairly low-rate but easy-to-use technology, NFC is also useful to set up more capable wireless connections such as Bluetooth.

NFC is covered by a number of standards starting from earlier ones on radio frequency identification (RFID): ISO/IEC 14443, FeliCa (by Sony), ECMA-340, ECMA-352, ISO/IEC 21481, and ISO/IEC 18092. The NFC Forum promotes implementation and standardization of NFC technology to ensure interoperability between devices and services [22].

In comparison with BLE, NFC has advantages of much lower cost and easier set up (versus pairing between BLE devices), but NFC suffers from a much shorter range and lower data rate.

NFC is an option for BLE out-of-band key exchange in addition to being a viable communication technology itself. Obviously, the short communication range is one natural challenge for eavesdroppers [26, 30]. The radio signal for wireless data transfer might be picked up less than 10 m, depending on multiple parameters. Also, passive devices are much harder to eavesdrop on than active devices, and an eavesdropper may have to be within a few centimeters.

However, plain NFC does not ensure secure communications and various attacks have been demonstrated. There is no protection against eavesdropping, data modification, or man-in-the-middle attacks. Applications use higher layer cryptographic protocols (e.g., SSL/TLS) for security.

4 Device Security

Wearables are vulnerable to attacks on hardware and software like any other computing devices.

4.1 System Security

Conventional desktop computers and operating systems such as Windows and Mac OS X are loaded with security features such as trusted platform module (TPM) chip, hard drive encryption, secure protocol suites (e.g., SSL/TLS and SSH), code signing, sandboxing, anti-malware software, and built-in firewalls. In comparison, wearable devices have much less computing, memory, and power resources, which impose serious limitations on feasible security features.

As mentioned earlier, wearables use embedded processors and SoCs. More security features are being implemented in these processors such as the TrustZone technology in the ARM Cortex-M23, Cortex-M33, and Cortex-A, and the secure enclave in Apple's A-series SoCs [7].

Wearable operating systems are a broad mixture of open-source (mostly based on Linux) and proprietary operating systems, with varying capabilities. Linux is a widely used operating system that is generally believed to be fairly secure. It is difficult to ascertain the security of proprietary operating systems.

Traditional cryptography poses a challenge for wearables. There is a recognized need for new lightweight cryptographic solutions with countermeasures to side-channel attacks that will be better for resource-constrained wearables [17].

4.2 Vulnerabilities

Verifying the firmware at update time is a step toward securing IoT devices; however, this is often done by the onboard software that is trusted to be authentic [7]. The implementation of this check must be sound. For example, schemes that utilize random numbers must ensure the usage of a cryptographically secure random number generator, and any used cryptographic certificates must be validated by a trusted certificate authority.

It may not be sufficient to just authenticate updates [7]. The software stack should also be authenticated; otherwise, the validity of an update cannot be determined reliably. Also, a proper chain of trust in the hardware architecture is needed before authenticating the software stack.

If a device is remotely updated, it must be able to check the integrity and authenticity of downloaded updates [7]. Typically, updates are protected cryptographically. However, errors and vulnerabilities have been seen in implementations.

Another point of vulnerability is debug interfaces [7]. Circuit board must expose programming interfaces and test points for testing the different components on the board. These interfaces are not removed after testing and might be used by adversaries to inject malicious code.

In sophisticated wearables capable of running different apps, there is a risk that apps might have vulnerabilities exploitable by adversaries. Since wearable apps are designed with tight hardware constraints, these apps can be inherently weaker than apps developed for desktop computing. For example, runtime bound checking might be eliminated to save computational power and memory space, thus exposing the apps to buffer overflow attacks.

In desktop computers, exploits might be caught and blocked by a host-based intrusion detection system (IDS). However, wearables do not have the computation and power resources to run intrusion detection [19].

A number of studies have experimentally looked for vulnerabilities in various commercial fitness trackers [28, 50, 70]. Most of the vulnerabilities found were related to the insecure implementation of communication protocols.

4.3 Malware

Much like desktop computers, wearables will be targets for malware [7]. Wearables are attractive targets because they hold a considerable amount of valuable information. Moreover, they are always connected to the network.

Linux has seen malware such as the Mirai bot. Some security companies anticipate an increase in Linux malware caused by an expansion of Internet of Things devices.

If wearables have any protection, it might consist of software level solutions such as firmware signing and code signing. Wearables do not have sufficient resources for traditional anti-malware software.

Hardware Trojans may also pose a threat. These are malicious modifications to integrated circuits that are difficult to detect by normal testing methodologies because they might be subtle. For example, a hardware Trojan inserted into a SoC might weaken the entropy of the random number generator used to generate keys. If these keys are used for encryption, the computational effort required by an adversary to decrypt data could be reduced greatly [7]. Hardware Trojans could require expensive specialized tests to detect them.

5 Privacy Issues

Most people think of privacy as the problem of data exposed to an eavesdropper, which is solved by encryption. However, data may be exposed in various ways. Privacy is a broader problem of a user controlling every aspect of where his or her personal data is represented, sent, stored, accessed, and possibly deleted.

Wearable devices including fitness trackers and medical devices are capable of collecting a variety of sensitive personal data. Therefore, they may be subject to privacy and regulatory policies such as the Health Insurance Portability and Accountability Act (HIPAA) that states the obligation for companies operating in the US to protect healthcare information [1].

Privacy issues are real for commercial wearable devices. An investigation of several wearable fitness trackers found a number of general privacy concerns [46].

5.1 Access Controls

Access control works in enforcing different access rights for different users. Data should be classified based on the sensitivity and each user will have different access levels. For example, a doctor will have more access rights than a nurse. Access control consists of authenticating the user, granting appropriate privileges, and revoking privileges. Due to their hardware constraints, the implementation of access control in wearables is still an open issue.

Examples of hardware implementations of protected data include ARM TrustZone, Apple's secure enclaves, and Samsung KNOX, as discussed earlier.

Access to data stored in the cloud must also be designed carefully. Homomorphic encryption has been proposed to ensure confidentiality of sensitive health data in the cloud [48]. Caregivers might be able to analyze the data which is unreadable to others, including the cloud service provider. However, homomorphic encryption is not practical for resource-constrained wearables.

5.2 Outsourcing

The current generation of wearables is much better than similar past devices (e.g., pedometers) in terms of their seamless integration with cloud services and online social networks. This integration raises security and privacy issues because by design, social networks are inherently open.

Unfortunately, wearables are too resource constrained to perform conventional methods to protect health data, e.g., de-identify data by data aggregation or removing common identifiers. An alternative is to move data to the cloud to take advantages of the computing and storage resources of cloud services. However, this approach introduces other privacy issues that have not been worked out entirely [70].

5.3 Health-Related Information of Non-health-Related Applications

Although most fitness trackers and wearable devices are not marketed as medical devices and therefore are not covered by health data protection regulations, they do store a considerable amount of user data that could be derived to extract health-related information. A study of BLE data traffic between a fitness tracker and a smartphone found a correlation to the intensity of the user's activity [18]. Experimental results with the Fitbit app and tracker showed that when the app was opened, the tracker would send a different amount of BLE packets depending on the activity the user was performing. This means that simply by observing and analyzing the encrypted BLE packets, an adversary could be able to guess the user's current activity (walking, sitting, and running).

5.4 Tracking

Wearable devices become particularly useful when they are connected to other devices. When wearable devices are interconnected, there could be a continuous exchange of data among them without being noticed by humans. In such a scenario, privacy may be easily breached (e.g., by revealing locations) [19].

This risk to privacy has been observed, for instance, in the Jawbone tracker. The BLE specification recommends that devices should change their Bluetooth device address frequently in order to prevent tracking, but this privacy feature is not implemented in the Jawbone tracker. It always uses the same address [28]. The static address allows the user to be tracked across visited locations.

In a similar way, Fitbit and other fitness trackers are constantly advertising themselves irrespective of whether they are already paired with some device or not

[18, 28]. In this case, the tracker uses the same device address and does not change it despite the BLE guidelines. Thus, a user might be tracked by listening to the Bluetooth traffic in an area.

6 Conclusions

Wearables are a diverse and expanding class of computing devices that pose many security and privacy issues, but our experience with them as a mass consumer device is limited to the past few years. The issues are more challenging for two major reasons. Wearables are designed to collect, store, and share a great deal of health data that might be considered personal or sensitive. At the same time, wearables have limited computation, memory, and power resources to implement a full suite of security features.

The increasing popularity of wearables among consumers is pushing commercial wearables into the marketplace before security and privacy issues can be worked out. This chapter has highlighted these issues because more research is needed to incorporate security features into wearables from the beginning of their design.

References

1. Al Alkeem, E., Yeun, C.Y., Zemerly, M.J.: Security and privacy framework for ubiquitous healthcare IoT devices. In: 10th International Conference for Internet Technology and Secured Transctions (ICITST), pp. 70–75. IEEE (2015)
2. Alliance, Z.: Zigbee alliance. http://www.zigbee.org
3. Amorim, V.J.P., Delabrida, S., Oliveira, R.A.R.: A constraint-driven assessment of operating systems for wearable devices. In: VI Brazilian Symposium on Computing Systems Engineering (SBESC), pp. 150–155. IEEE (2016)
4. Android: Android wear. https://www.android.com/intl/en_uk/wear/
5. ANT: The wireless sensor network solution this is ant. https://www.thisisant.com
6. Aoyagi, T., Miyasaka, K.: Pulse oximetry: its invention, contribution to medicine, and future tasks. Anesth. Analg. **94**(1 Suppl), S1 (2002)
7. Arias, O., Wurm, J., Hoang, K., Jin, Y.: Privacy and security in Internet of Things and wearable devices. IEEE Trans. Multi-Scale Comput. Syst. **1**(2), 99–109 (2015)
8. ARMmbed: mbed os. https://www.mbed.com/en/platform/mbed-os/
9. Baldini, F.: Invasive sensors in medicine. In: Optical Chemical Sensors, pp. 417–435. Springer (2006)
10. Blasco, J., Chen, T.M., Tapiador, J., Peris-Lopez, P.: A survey of wearable biometric recognition systems. ACM Comput. Surv. (CSUR) **49**(3), 43:1–35 (2016)
11. Boll, S., Pulsipher, D.: Suppression of acoustic noise in speech using two microphone adaptive noise cancellation. IEEE Trans. Acoust. Speech Signal Process. **28**(6), 752–753 (1980)
12. Bouhenguel, R., Mahgoub, I., Ilyas, M.: Bluetooth security in wearable computing applications. In: International Symposium on High Capacity Optical Networks and Enabling Technologies, pp. 182–186. IEEE (2008)
13. Braasch, M.S., Van Dierendonck, A.J.: Gps receiver architectures and measurements. Proc. IEEE **87**(1), 48–64 (1999)

14. Catalano, J.T.: Guide to ECG analysis. Lippincott Williams & Wilkins (2002)
15. Chatterjee, A., Aceves, A., Dungca, R., Flores, H., Giddens, K.: Classification of wearable computing: a survey of electronic assistive technology and future design. In: 2nd International Conference on Research in Computational Intellegence and Communication Networks (ICRCICN), pp. 22–27. IEEE (2017)
16. Cornacchia, M., Ozcan, K., Zheng, Y., Velipasalar, S.: A survey on activity detection and classification using wearable sensors. IEEE Sens. J. **17**(2), 386–403 (2017)
17. Cruz, R.J., Reis, T.B., Aranha, D.F., Kupwade Patil, H.: Lightweight cryptography on arm. In: NIST Lightweight Cryptography Workshop. NIST (2016)
18. Das, A.K., Pathak, P.H., Chuah, C.N., Mohapatra, P.: Uncovering privacy leakage in ble network traffic of wearable fitness trackers. In: 17th International Workshop on Mobile Computing Systems and Application, HotMobile '16, pp. 99–104. ACM (2016)
19. Di Pietro, R., Mancini, L.V.: Security and privacy issues of handheld and wearable wireless devices. Commun. ACM **46**(9), 74–79 (2003)
20. Diffie, W., Hellman, M.: New directions in cryptography. IEEE Trans. Inf. Theory **22**(6), 644–654 (1976)
21. Dunning, J.P.: Taming the blue beast: a survey of bluetooth based threats. IEEE Secur. Priv. **8**(2), 20–27 (2010)
22. Forum, N.: Nfc forum. http://nfc-forum.org
23. Gao, W., Emaminejad, S., Nyein, H.Y.Y., Challa, S., Chen, K., Peck, A., Fahad, H.M., Ota, H., Shiraki, H., Kiriya, D.: Fully integrated wearable sensor arrays for multiplexed in situ perspiration analysis. Nature **529**(7587), 509–514 (2016)
24. Gartner: Gartner says worldwide wearable devices sales to grow 18.4 percent in 2016 (2016). http://www.gartner.com/newsroom/id/3198018
25. Ghamari, M., Arora, H., Sherratt, R.S., Harwin, W.: Comparison of low-power wireless communication technologies for wearable health-monitoring applications. In: 2015 International Conference on Computer, Communication, and Control Technology (I4CT), pp. 1–6. IEEE (2015)
26. Ghosh, S., Goswami, J., Kumar, A., Majumder, A.: Issues in NFC as a form of contactless communication: a comprehensive survey. In: International Conference on Smart Technology and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), pp. 245–252. IEEE (2015)
27. Gomez, C., Oller, J., Paradells, J.: Overview and evaluation of bluetooth low energy: an emerging low-power wireless technology. Sensors **12**(9), 11734–11753 (2012)
28. Goyal, R., Dragoni, N., Spognardi, A.: Mind the tracker you wear: a security analysis of wearable health trackers. In: 31st Annual ACM Symposium on Applied Computing (SAC '16), pp. 131–136. ACM (2016)
29. Gupta, N.K.: Inside Bluetooth Low Energy. Artech House (2016)
30. Haselsteiner, E., Breitfuß, K.: Security in near field communication (NFC). In: Workshop on RFID security, pp. 12–14 (2006)
31. Islam, N., Want, R.: Smartphones: past, present, and future. IEEE Pervasive Comput. **13**(4), 89–92 (2014)
32. Islam, S.K., Fathy, A., Wang, Y., Kuhn, M., Mahfouz, M.: Hassle-free vitals. IEEE Microw. Mag. **15**(7), S25–S33 (2014)
33. Jiang, H., Chen, X., Zhang, S., Zhang, X., Kong, W., Zhang, T.: Software for wearable devices: challenges and opportunities. In: IEEE 39th Annual Computer Software and Applications Conference, pp. 592–597. IEEE (2015)
34. Ko, M., Goeckel, D.L.: Wireless physical-layer security performance of UWB systems. In: IEEE MILCOM 2010, pp. 2143–2148. IEEE (2010)
35. Koblitz, N.: Elliptic curve cryptosystems. Math. Comput. **48**, 203–209 (1987)
36. Labrador, M.A., Yejas, O.D.L.: Human Activity Recognition Using Wearable Sensors and Smartphones. Taylor and Francis Group, Boca Raton, FL (2014)
37. Labs, M.: Linkit assist 2502. https://labs.mediatek.com/en/platform/linkit-assist-2502

38. Lara, O.D., Labrador, M.A.: A survey on human activity recognition using wearable sensors. IEEE Commun. Surv. Tutor. **15**(3), 1192–1209 (2013)
39. Ltd, A.: Arm processors. http://www.arm.com/products/processors
40. Ltd, A.: A system-wide approach to security. https://www.arm.com/products/security-on-arm/trustzone
41. Moody, G., Mark, R., Bump, M., Weinstein, J., Berman, A., Mietus, J., Goldberger, A.: Clinical validation of ecg-derived respiration (edr) technique. Comput. Cardiol. **13**, 507–510 (1986)
42. Mosenia, A., Sur-Kolay, S., Raghunathan, A., Jha, N.K.: Wearable medical sensor-based system design: a survey. IEEE Trans. Multi-Scale Comput. Syst. **PP**(99), 1–1 (2017)
43. Moyer, J., Wilson, D., Finkelstein, I., Wong, B., Potts, R.: Correlation between sweat glucose and blood glucose in subjects with diabetes. Diabetes Technol. Ther. **14**(5), 398–402 (2012)
44. Nourbakhsh, N., Wang, Y., Chen, F., Calvo, R.A.: Using galvanic skin response for cognitive load measurement in arithmetic and reading tasks. In: 24th Australian Computer-Human Interaction Conference, pp. 420–423. ACM (2012)
45. Pantelopoulos, A., Bourbaki, N.G.: A survey on wearable sensor-based systems for health monitoring and prognosis. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **40**(1), 1–12 (2010)
46. Paul, G., Irvine, J.: Privacy implications of wearable health devices. In: 7th International Conference on Security of Infomation and Networks (SIN ’14), pp. 117:117–121. ACM (2014)
47. Perera, C., Liu, C.H., Jayawardena, S.: The emerging Internet of Things marketplace from an industrial perspective: a survey. IEEE Trans. Emerg. Topics Comput. **3**(4), 585–598 (2015)
48. Preuveneers, D., Joosen, W.: Privacy-enabled remote health monitoring applications for resource constrained wearable devices. In: 31st Annual ACM Symposium on Applied Computing (SAC ’16), pp. 119–124. ACM (2016)
49. Qualcomm: Snapdragon wear 2100 processor product brief. https://www.qualcomm.com/documents/snapdragon-wear-2100-processor-product-brief
50. Rahman, M., Carbunar, B., Topkara, U.: Secure management of low power fitness trackers. IEEE Trans. Mob. Comput. **15**(2), 447–459 (2016)
51. Ray, S., Park, J., Bhunia, S.: Wearables, implants, and Internet of Things: the technology needs in the evolving landscap. IEEE Trans. Multi-Scale Comput. Syst. **2**(2), 123–128 (2016)
52. Roggen, D., Perez, D.G., Fukumoto, M., van Laerhoven, K.: Iswc 2013—wearables are here to stay. IEEE Pervasive Comput. **13**(1), 14–18 (2014)
53. Roman, R., Najera, P., Lopez, J.: Securing the Internet of Things. IEEE Comput. **44**(9), 51–58 (2011)
54. Romano, S.M., Pistolesi, M.: Assessment of cardiac output from systemic arterial pressure in humans. Crit. Care Med. **30**(8), 1834–1841 (2002)
55. RoweBots: Wearableos. https://rowebots.com/en/products/wearableos-unison-wearable-operating-system
56. Ryan, M., et al.: Bluetooth: with low energy comes low security. In: WOOT (2013)
57. Salvo, P., Francesco, F.D., Constanzo, D., Ferrari, C., Trivella, M.G., Rossi, D.D.: A wearable sensor for measuring sweat rate. IEEE Sens. J. **10**(10), 1557–1558 (2010)
58. Samsung: Exynos 9 series (8895). http://www.samsung.com/semiconductor/minisite/Exynos/w/solution/mod_ap/8895/
59. Sazonov, E., Neuman, M.R.: Wearable Sensors: Fundamentals. Academic Press, Implementation and Applications (2014)
60. SIG, B.: Bluetooth technology website. https://www.bluetooth.com
61. Starner, T.: The challenges of wearable computing: Part 1. IEEE Micro **21**(4), 44–52 (2001)
62. Starner, T.: How wearables worked their way into the mainstream. IEEE Pervasive Comput. **13**(4), 10–15 (2014)
63. Statista: Wearable device shipments worldwide from 2015 to 2021 (in million units). https://www.statista.com/statistics/610478/wearable-device-shipments-worldwide/
64. Tamura, T., Maeda, Y., Sekine, M., Yoshida, M.: Wearable photoplethysmographic sensors—past and present. Electronics **3**(2), 282–302 (2014)
65. Tizen: Tizen. https://www.tizen.org

66. Vashist, S.K.: Non-invasive glucose monitoring technology in diabetes management: a review. Anal. Chim. Acta **750**, 16–27 (2012)
67. Wang, S., Bie, R., Zhao, F., Zhang, N., Cheng, X., Choi, H.A.: Security in wearable communications. IEEE Netw. **30**(5), 61–67 (2016)
68. Whiting, D., Housley, R., Ferguson, N.: Counter with cbc-mac (ccm). In: RFC 3610. IETF (2003). http://www.ietf.org/rfc/rfc3610.txt
69. Zhang, M., Raghunathan, A., Jha, N.K.: Trustworthiness of medical devices and body area networks. Proc. IEEE **102**(8), 1174–1188 (2014)
70. Zhou, W., Piramuthu, S.: Security/privacy of wearable fitness tracking iot devices. In: 9th Iberian Conference on Infomation Systems and Technologies (CISTI) (2014)
71. Zieniewicz, M.J., Johnson, D.C., Wong, D.C., Flatt, J.D.: The evolution of army wearable computers. IEEE Pervasive Comput. **1**(4), 30–40 (2002)