Smartwatches and Fitness Trackers: Cyberphysical Privacy and Security Threats

Henrik Strangalies
Freie Universität Berlin
IoT & Security Seminar Report

***Abstract*—Wearable devices have become increasingly popular due to their convenience and functionality, enabling users to perform various tasks such as making payments, monitoring health, and receiving messages. However, along with these benefits, wearables bring forth security and privacy concerns. This report aims to explore the challenges associated with wearables, emphasizing the importance of robust security measures and privacy safeguards. Specifically, it delves into the security and privacy threats posed by accelerometer and gyroscope data. Subsequently, the report focuses on a research paper that examines the risks of analyzing motion sensor data to decipher keyboard inputs being able to infer typed words on the keyboard. The findings from these papers highlight the vulnerabilities in wearable technologies and underscore the need for effective security measures to mitigate these threats.**

***Index Terms*—Wearables, Security, Privacy.**

I. Introduction

The rising popularity of wearable devices stems from their convenience and functionality, allowing users to perform tasks like payments, health monitoring, and message reception. Therefore, the report deals first with the IoT devices, smartwatches and fitness trackers. Nonetheless, the advantages of wearables are accompanied by security and privacy challenges. Particularly, the study [8] found that most users had limited awareness of the risks associated with fitness trackers and displayed a lack of concern. They also showed a tendency to neglect adjusting security settings or adopting additional measures. Another study [9] explores how users perceive the trade-off between privacy and utility when using fitness trackers. It discusses participants' attitudes towards data collection, sharing practices, and the benefits of using fitness trackers. It has been found that participants were aware that certain types of information could be inferred from the data collected by fitness trackers. For example, they correctly recognized that heart-rate data could reveal sexual activity. However, they were unaware that non-physiological information could also be inferred from the data.

Given the numerous advantages of wearables and users' limited understanding of security and privacy risks, there is now a growing focus on addressing this issue appropriately. Therefore, the paper [1] provides a comprehensive overview of wearable devices (including smart watches, wrist bands, smart glasses, smart jewellery, electronic garments and skin patches) and the associated challenges. The survey covers various aspects of wearables, including their architecture, communication protocols, energy efficiency, security, privacy, and applications. The paper also highlights the existing challenges such as limited battery life, data privacy concerns, user interface design, and interoperability issues. Furthermore, this report covers also privacy threats using the papers [2] and [5]. The paper [2] provides an extensive overview of the various sensors found in smartphones and tablets, including cameras, microphones, accelerometers, gyroscopes, and GPS. The authors discuss the potential privacy risks arising from sensor data collection and usage, such as location tracking, activity tracking, biometric identification, and behavioral profiling. The paper [5] examines the privacy implications arising from accelerometer data collected by various devices, including smartphones, wearables, and IoT devices. The authors discuss the potential inferences that can be made from accelerometer data, such as activity recognition, behavior profiling, gait analysis, and biometric identification. They analyze the privacy risks associated with these inferences and discuss the existing privacy protection mechanisms and regulations. The review provides valuable insights into the privacy considerations and challenges associated with accelerometer data and calls for further research to address these concerns effectively.

Specifically, this report delves into the security and privacy threats posed by accelerometer and gyroscope data collected by wearables. It focuses on the research paper [4] that examines the risks of analyzing motion sensor data to decipher keyboard inputs. Keyboards are widely used for inputting sensitive information, such as PINs and confidential documents, making them an attractive target for cybercriminals. While keyloggers can directly steal keystrokes, they require storage on the victim's machine. In contrast, side-channel based keystroke inference attacks gather signals emitted from keyboards using external devices, providing a stealthier approach. Numerous studies have investigated potential channels for these types of attacks. The findings from these papers highlight the vulnerabilities in wearable technologies and underscore the need for effective security measures to mitigate these threats.

II. IoT wearable hardware: Smartwatches and Fitness Tracker

Smartwatches have become one of the most popular types of wearable devices. According to recent reports, smartwatch sales rank second in the wearables market, with approximately 50 million units projected to be sold in 2016 [1].

Smartwatches, such as the Samsung Galaxy Gear and Apple Watch, are equipped with modern operating systems like

Android Wear and Watch OS, respectively, and optimized for smartwatches with limited processing power and battery life. These operating systems allow users to install various applications that provide advanced functionalities, including making phone calls and checking messages. Additionally, smartwatches are equipped with multiple sensors, such as accelerometers, microphones, gyroscopes, and heart rate sensors, which gather information about the user and the surrounding environment [4].

The functionality of smartwatches typically serves two main purposes. Firstly, they act as communication and notification tools, providing users with convenient access to various smartphone features. This includes receiving notifications such as phone calls, text messages (SMS), emails, voice control functionalities, and weather updates. Additionally, smartwatches allow for micro interactions, enabling users to perform tasks like launching smartphone apps, limited web browsing, setting reminders, and issuing voice commands [1].

Secondly, many smartwatches are equipped with sensors that can monitor human physiological signals and biomechanics. This allows them to function as fitness tracking devices, enabling users to log their daily activities. For example, smartwatches can automatically record workout times, track heart rate, count steps, and estimate calories burned. The collected data is then transferred to a smartphone or cloud server for further analysis and presentation to users, often through interactive dashboards [1].

The importance of energy in wearable computing cannot be overstated, especially as user demands for enhanced functionalities and power increase. Frequent recharging of wearable devices by plugging them into power sources is often inconvenient for many users. Wireless charging has emerged as a more flexible and convenient solution. However, it tends to have longer charging times due to limited energy transfer rates. To achieve energy efficiency in wearables, two general approaches are commonly pursued: advancing battery technology and minimizing power consumption. Battery technology has made linear progress over the years compared to the exponential growth seen in electronic devices, but advancements in materials, architectures, surface structures, and chemical reactions can further enhance battery technology for wearable devices. Energy harvesting has also gained attention from both research and industrial communities as a means of powering wearables. Moreover, reducing power consumption through energy-efficient solutions is an active research area that can prolong the battery life of wearables. However, wearables still require recharging or battery replacement at some point. For example, the recent Apple Watch offers only 18 hours of operation before needing to be recharged daily. This frequent need for recharging diminishes the desirability of wearables, especially in health applications such as aged-care, where users may easily forget to recharge their devices. Therefore, the development of self-powered wearables using energy harvesting technology is highly desirable. Energy harvesting is a rapidly growing field that attracts significant efforts from both the research community and industry [1].

Research directions aimed at addressing the computing challenges faced by wearables can be categorized into three main approaches, each with its own advantages and disadvantages. The first approach is hardware offloading, which involves moving computationally intensive tasks to specialized hardware. While this approach offers low latency and high accuracy for specific tasks, it may increase the device's footprint and may not be suitable for all features offered by wearables. Future trends in the smartphone domain suggest the possibility of wearables having dedicated hardware for machine learning and artificial intelligence functions like face and speech recognition. The second approach is cloud offloading, which leverages the computational power of the cloud to scale up the capabilities of wearables. However, this approach comes with increased latency and communication costs. The third approach is in-device machine learning, which strikes a balance between specialized hardware and utilizing the full potential of the cloud. It involves using the wearable device's CPU and GPU to implement more powerful machine learning models, including state-of-the-art deep learning models, to achieve high accuracy. In many cases, researchers aim to import compact versions of high-performance machine learning models or select specific layers to optimize accuracy based on the wearable device's hardware constraints. Recent research has even proposed dynamic partitioning of neural network models between the cloud and the wearable device, resulting in low latency and energy consumption [1].

Additionally, wireless communication protocols are essential for the functionality of wearables, allowing them to transmit information for various applications. However, because wearables often handle sensitive data, such as health and financial information, they must prioritize security to protect against potential risks associated with wireless data access. However, resource limitations in wearables, including limited battery life, CPU power, memory, and device form factor, often restrict the implementation of advanced security measures. Therefore, it is crucial to identify security vulnerabilities in wearable wireless communications and develop practical safeguards to address them [1].

Wristbands are a popular category of wearable devices that focus on health and fitness tracking. They often lack display screens and have a more compact form factor compared to smartwatches. Wristbands can passively track and record user activities like walking, running, and sleep. They typically include sensors such as bio-impedance sensors to measure heart rate, tri-axis accelerometers, and temperature sensors. Wristbands like the UP4 utilize smartphone apps for data visualization, and they may also offer additional features such as NFC for making payments similar to a credit card [1].

III. Panorama of Security & Privacy Considerations with IoT wearables

The survey of Seneviratne, Suranga et al. [1] enlightens security threats to wearables under the three categories: Threats to confidentiality, integrity, and availability. Therefore, this

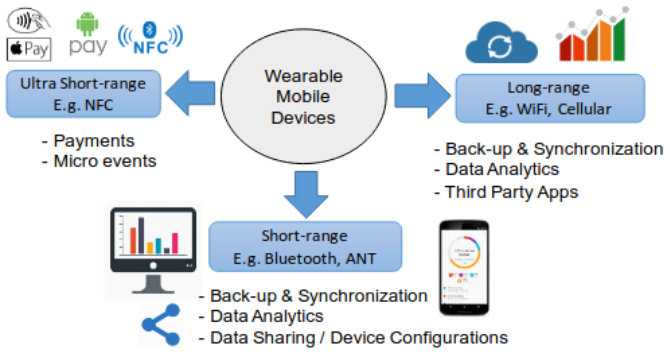

Fig. 1. Communication modes of wearable devices. The figure is taken from [1].

section is divided into these categories. Threats to Confidentiality encompasses those where attackers get unauthorised access to information using techniques such as eavesdropping the wireless channel. Threats to Integrity includes the cases where attackers alter data or information without authorisation. Threats to Availability are the situations where attackers act to deny services to the entities who are authorised to use them [1].

A. Threats to Confidentiality

The survey [1] pointed out that most existing wearable devices use Bluetooth Low Energy (BLE) as the major means of communication. However, it has been shown that BLE equipped wearable devices are prone to attacks that impact the confidentiality such as eavesdropping and traffic analysis. Another vulnerability is where information about user's other devices such as the PIN to unlock the smartphone is gathered using wearable devices. In the following existing attacks threatening confidentiality in wearable communications are discussed.

1) Eavesdropping Attacks: Eavesdropping is the unauthorized real-time interception of a private communication which can expose user's personal information to an attacker. Particularly, wearable devices using BLE communication protocol can suffer from eavesdropping. It has been found out that some of the existing wearables do not have a proper implementation of the MAC address randomisation defined in the BLE specification as a privacy preserving provision. For instance, it has been discovered that some devices do not use address randomisation at all whilst some implementations only flipping few bits of the public addresses and others raised concerns in the duration a random address is kept active. These incautious implementations enable adversaries to eavesdrop on the wireless channel and follow the advertisement packets to track a BLE device easily. For example, Goyalet et al. [6] identified the key privacy and security vulnerabilities of the two most widely adopted wearable fitness trackers, Jawbone UP Move and FitBit Charge and discovered that both trackers have not implemented address randomisation. Thus, the attacker can track the user over time by eavesdropping on the device's communications and exploiting the public device address. Similarly, the authors of the Open Effect Report from 2016 [7] investigated BLE privacy provision in number of fitness tracking devices such as Fitbit Charge HR, Jawbone UP 2, Garmin Vivosmart, Apple Watch, and Xiaomi Mi Band and came to the conclusion all tested devices, except Apple Watch, use the static device addresses that allowed attackers to track user information such as location, time of fitness activities, and reversing user profile by eavesdropping on these devices' communications [1].

2) Traffic Analysis Attacks: Traffic analysis attacks in the context of wearables involve monitoring communication patterns between devices. Privacy vulnerabilities have been identified in Bluetooth Low Energy (BLE) communication between fitness trackers and smartphones. Adversaries can track users by analyzing BLE advertisements and static device addresses. Additionally, user activities can be inferred from the size and number of data packets in BLE traffic, even if the packets are encrypted. Unique walking patterns can also be used to identify individuals within a small group, even with random addresses [1].

In the paper [15], a measurement-driven study is presented to examine possible privacy leakage from BLE communication between the fitness tracker and the smartphone. By utilizing real BLE traffic traces collected in the wild and controlled experiments, it has been shown that the majority of fitness trackers use unchanged BLE addresses during advertising, making it feasible to track them. The BLE traffic of the fitness trackers is found to be correlated with the intensity of the user's activity, enabling an eavesdropper to determine the user's current activity (walking, sitting, idle, or running) through analysis of the BLE traffic. Furthermore, it is also demonstrated that the BLE traffic can represent the user's gait, which is known to be distinct from user to user. This makes it possible to identify a person (from a small group of users) based on the BLE traffic emitted by their fitness tracker. As BLE-based wearable fitness trackers become widely adopted, the aim of this study is to identify important privacy implications of their usage and discuss prevention strategies.

3) Information Gathering Attacks: Passive monitoring of wearable device transmissions enables adversaries to collect data exchanged between wearables and their hubs. This information can be used for information gathering attacks, including breaking key exchanges in Bluetooth Low Energy (BLE) pairing and gathering information about user's other devices. Researchers have demonstrated attacks that break BLE legacy pairing, infer keystrokes on smartphone touchpads using smartwatch motion sensors, decode keystrokes on keyboards using smartwatch sensors, and infer a user's personal PIN sequence using wearable devices. Adversaries can gain access to smartwatches by installing malicious applications to record sensor activities. These attacks leverage sensor data captured by wearables and can be executed by sniffing BLE communications or installing malicious apps on wearables [1].

4) Other Attacks: A limited number of studies have focused on the vulnerabilities of wearable systems and associated

applications. One analysis examined the security aspects of Android apps from Jawbone and Fitbit. It was found that the Jawbone app shared device information with third parties, while both apps stored preferences and data in plaintext, making them accessible to attackers. Another study by HP revealed that many consumer smartwatches lacked critical security measures such as data encryption and user authentication. These vulnerabilities stem from inadequate implementation by manufacturers, who prioritize quick product launches over security. Future generations of wearable products are expected to address these issues, but the resource-security trade-off will remain. Effective countermeasures to protect the confidentiality of wearable communications are crucial [1].

B. Threats to Integrity

Integrity is a crucial security requirement for wearable systems, particularly due to the sensitivity and privacy of the collected data. Ensuring that data remains unaltered during transmission and reaches only authorized parties is paramount. Various studies in the literature have evaluated the integrity of wearable device systems, identifying vulnerabilities in three attack categories: Modification Attacks, Replay Attacks, and Masquerade Attacks [1]. Additionally, this section comprises an overview of possible data breaches.

1) Modification Attacks: In wireless data transmission between wearable devices, there is a risk of data modification or alteration. Adversaries can intercept and modify data exchanged between wearable devices, including changing packet content and timestamps. Vulnerabilities have been found in Bluetooth LE pairing, fitness data storage, and transmission in popular trackers such as FitBit and Garmin. Attackers can exploit these vulnerabilities to capture, modify, and inject data. Timestamp integrity in healthcare devices has also been compromised, allowing attackers to tamper with medical data. Furthermore, the lack of HTTPS transmission in certain applications exposes sensitive fitness data to unauthorized parties, enabling data falsification [1].

2) Replay Attacks: In replay attacks, adversaries capture valid data packets uploaded by a wearable device and replay them for malicious purposes such as impersonation or data corruption. An example for a replay attack was demonstrated in a commercially available insulin delivery system where by eavesdropping on the communication between devices, attackers gathered information transmitted in plaintext, including device type, PIN, therapy or glucose level, and patient's medical condition. Through brute-force methods, they determined the Cyclic Redundant Check parameters used in the system and performed replay attacks by altering the counter field of the packet, reporting outdated glucose levels as an example [1].

3) Masquerade Attacks: Masquerade attacks involve impersonating an authenticated device to steal data or inject fake information. Examples include collecting bonding information from medical devices through malicious apps and controlling insulin pumps by knowing the device's PIN. These attacks exploit the lack of authentication and encryption in wearable systems. While threats to integrity are less common than those to confidentiality, addressing data confidentiality vulnerabilities will also protect data integrity [1].

4) Data Breaches: Smartwatches and fitness tracker bracelets can provide measurements such as distance walked or run using motion sensors and GPS. They can also monitor physiological/biological parameters like heart rate, ECG, stress levels, sleep quality, and more. The mobile applications provide a comprehensive user interface to visualize, analyze, and manage this data, offering users a holistic view of their health and activity information. Furthermore, optical sensors are commonly integrated into smartphones and smartwatches to detect variations in blood volume within the arteries beneath the skin. This enables the measurement of heart-related metrics and polysomnographic parameters. Indeed, data collection is paramount for providing users with plenty features to track theirs health and lead to a healthy lifestyle. On the other hand, the massive data collection is prune for data breaches. Furthermore, it has been shown that the identification of various activities such as stationary behavior, walking, running, bicycling, stair climbing, descent, and driving was achieved solely through the utilization of accelerometer data. Information pertaining to sleep, including sleep posture and habits, has been successfully extracted using motion sensors. A study employed accelerometer, gyroscope, and orientation data from a smartwatch to detect sleep posture, achieving an accuracy exceeding 95% by utilizing the Euclidean distance of the input values. Additionally, the same study utilized these sensors to detect the hand position during sleep, achieving an accuracy of over 88% through the application of the k-NN algorithm [2].

C. Availability

Denial of Service (DoS) attacks are a common type of attack against availability in wearable devices. They aim to disrupt communication between wearables and their base or overwhelm the device's storage capacity with useless information. Examples of such attacks have been documented in the research literature. For instance, the FitBit Charge tracker can be targeted with DoS attacks to prevent legitimate device syncing or pairing with mobile phones. Attack tools like Fitbite and GarMax have been used to inject fake data, exceeding the storage capacity of trackers, thereby preventing them from recording valid user data. Additionally, these attacks can drain the battery by continuously querying the nearby trackers. It is important for manufacturers to address these implementation shortcomings in future wearable products, especially in critical healthcare devices like insulin pumps that require uninterrupted functionality. Continuous vigilance and vulnerability assessments are crucial to ensure the availability of wearable systems [1].

IV. Threats to Security and Privacy from Accelerometer Data

Particularly, privacy and security risks arise from the collection and utilization of accelerometer data on smartwatches and fitness trackers. The accelerometer, which measures motion

and movement, can inadvertently disclose sensitive information about an individual's activities and behavior. This data, if not properly protected, can be accessed by unauthorized parties and potentially lead to privacy or security breaches. Safeguarding the privacy of accelerometer data is crucial to ensure the confidentiality and security of users' personal information. The paper by Kröger et al. [5] is predestined for exemplifying that topic, since it highlights the potential privacy implications of accelerometers and security aspects such as inferring passwords, which are commonly found in mobile devices. While accelerometers are generally considered non-intrusive and do not require special permissions, research has shown that they can be used as a side channel to infer sensitive information about device holders. Accelerometer data alone can reveal details such as location, activities, health condition, body features, gender, age, personality traits, and emotional state. It can even be used for biometric identification and reconstructing text sequences, including passwords. Given these findings, the paper suggests that accelerometers should be re-evaluated in terms of privacy implications, and corresponding adjustments should be made to sensor protection mechanisms.

Furthermore, this work presents the report of the paper [3]. Here, the significant ramifications of data leakage resulting from the camouflage of smartwatches as activity trackers are highlighted in this paper, as it compromises the privacy of emails, search queries, and other keyboard-typed documents of the user. In view of the importance of addressing privacy and security risks pertaining to accelerometer data, this paper has been chosen for subsequent investigation and analysis.

A. Activity And Behavior Tracking

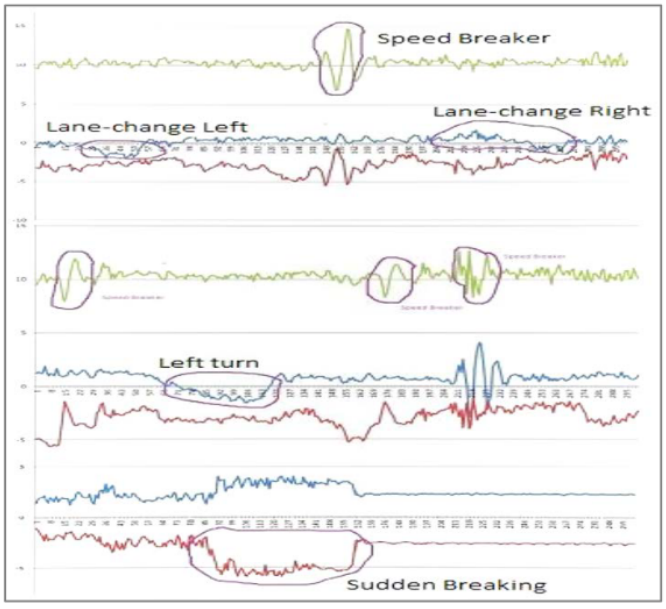

Fig. 2. Classification of driving patterns based on streams of accelerometer data. The Fig. is taken from [5].

Accelerometers provide valuable data for a wide range of applications and can derive various physical activity variables and behavior-related information. They are used in step counters to estimate energy expenditure and distance walked, and in medical studies to assess sedentary time and physical activity. Accelerometers enable real-time body posture and activity classification, including basic activities like running, walking, and sitting, as well as more complex activities like writing, typing, and painting. They can also monitor sleep patterns and behaviors. Additionally, accelerometers can detect hand gestures, eating and drinking moments, smoking, and even distinguish levels of intoxication. They have been used to detect carried loads and estimate carried weight, measure driving behavior, analyze speech activity and social interactions, and reconstruct speech from recorded vibrations. In Fig. 2, it can be observed that researchers were able to detect aggressive or unsafe driving styles and drunk driving patterns. The potential applications of accelerometers are vast and extend to various domains such as healthcare, fitness tracking, and behavior analysis [5].

B. Location Tracking

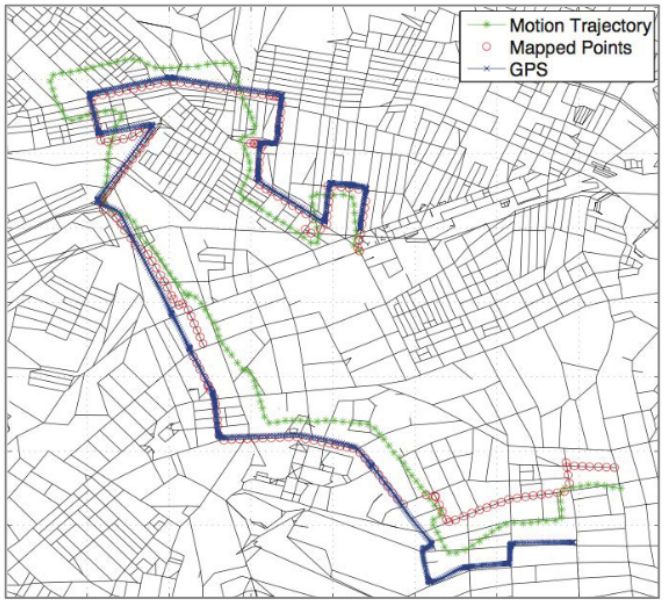

Fig. 3. **Map matching algorithm.** The green trail indicates the motion trajectory obtained from accelerometer data. The red trail indicates the inferred route. The blue trail indicates the actual route traveled (GPS data). The Fig. is taken from [5].

Studies have demonstrated that accelerometers in mobile devices can be utilized for user localization and reconstruction of travel trajectories, even in the absence of GPS or other localization systems. Researchers have achieved geographically tracking individuals driving a car solely based on accelerometer readings from their smartphones. By analyzing three-axis acceleration measurements and mapping them to existing routes on a map, they obtained trajectory information with accuracy comparable to handheld GPS devices. An example is illustrated in Fig. 3 of that algorithm. Another study focused on using smartphone accelerometers to determine the location

of the device user within a metropolitan train system. By comparing acceleration patterns with labeled training data, specific station intervals were recognized, and the accuracy of the approach reached up to 89% for rides longer than 3 stations and 92% for rides longer than 5 stations. These findings highlight the potential privacy risks associated with accelerometer data and the ability to infer sensitive information about individuals' whereabouts and travel patterns [5].

C. User Identification

Accelerometers in mobile devices have demonstrated the ability to differentiate between and uniquely identify users based on their body movement patterns. Biometric features such as gait, hand gestures, and head movements recorded by accelerometers have been used for user identification with high accuracy. For example, one study achieved 100% accuracy in recognizing individuals from a test group using accelerometer readings from smartphones. Additionally, accelerometers can capture sound vibrations, including human speech, with enough quality to distinguish between different speakers accurately.

The trajectory of a mobile device, inferred from accelerometer data, can reveal a user's work and home addresses. When combined with other auxiliary datasets, such as white pages or employment directories, it can potentially expose a user's real identity. Device fingerprinting techniques can further differentiate users based on unique characteristics of their personal devices. Calibration errors in accelerometers, caused by manufacturing imperfections, have been found sufficient to create a device "fingerprint" that can track users across website visits, even when other tracking technologies like cookies are blocked [5].

D. Keystroke Logging

The input that users type into their devices, whether through touchscreens or keyboards, often contains highly sensitive information such as text messages, personal notes, login credentials, and transaction details. Researchers have shown that motion sensor data, including accelerometer readings, can be used to reconstruct user inputs from handheld and wrist-worn devices. By analyzing the hand movements associated with swipes, taps, and keystrokes, researchers have successfully inferred inputs from motion sensor data. Some studies have exclusively relied on accelerometer data for keystroke inference attacks. For instance, researchers have demonstrated that accelerometers in smartphones can be exploited to infer tap- and gesture-based inputs, including PINs and graphical password patterns. Additionally, entire sequences of text entered through a phone's touchscreen have been obtained using accelerometer data. The fact that even multi-sensor attacks predominantly use acceleration information for tap detection highlights the importance of focusing on defense mechanisms against these types of side-channel attacks, particularly in relation to accelerometers [5].

In the paper of Liu et al. [4], the security problem of breaking passwords and PINs arising from smartwatch sensors is investigated. The accelerometer integrated into a smartwatch, being worn on the wrist, enables the tracking of user hand movements, thereby theoretically facilitating the inference of user inputs on keyboards. However, the practical implementation of such inference encounters several challenges in real-world settings. Persistent occurrences of small and irregular hand movements during typing degrade tracking accuracy and often overshadow useful signals. A new and practical side-channel attack is presented in this paper to infer user inputs on keyboards by exploiting smartwatch sensors. The approach involves the development of novel keystroke inference models to mitigate the adverse effects of tracking noises. The study focuses on two major keyboard categories: numeric keypads commonly used for digit input and QWERTY keyboards for English text entry. Two prototypes are built to infer users' banking PINs during typing on point-of-sale (POS) terminals and English text on QWERTY keyboards, respectively. The results indicate a high probability (up to 65%) of identifying banking PINs within the top three candidates for numeric keyboards. Furthermore, significant accuracy improvements are achieved for QWERTY keyboards compared to previous works, particularly in terms of the success rate of identifying the correct word within the top ten candidates. However, the transformation of stream data into accurate keystrokes is not a straightforward process. Motion sensors can capture hand movements, but several challenges need to be addressed before utilizing the data effectively. Firstly, there exists a significant variance in hand movements. For example, the speed at which the hand moves is not constant across different keys. Individuals tend to type faster for familiar words and slower for unfamiliar ones. Secondly, the collected data is prone to noise. Small and irregular hand movements consistently occur during key presses, leading to instability in the input data [4].

The paper of Sarkisyan et al. [13] proprosed an approach where in a controlled scenario, PIN prediction accuracy of no less than 41% and up to 92% within five attempts is achieved. This study presents a method for inferring smartphone PINs through the analysis of motion sensors in a smartwatch. The primary objective of this investigation is to assess the feasibility and accuracy of inferring user keystrokes on a smartphone using a smartwatch worn by the user. Specifically, the study demonstrates that user activity and specific numeric keypad entries can be recognized by accessing only the motion sensors of the smartwatch through malware.

In the paper of Nerini [14], it has been shown that information about smartphone movements can lead to the identification of a Personal Identification Number (PIN) typed by the user. To reduce the amount of sniffed data, an event-driven approach is used, where motion sensors are sampled only when a key is pressed. The acquired data are used to train a Machine Learning model for the classification of the keystrokes in a supervised manner. It is also considered that users insert the same PIN each time authentication is required, leading to further side-channel information available to the attacker. Numerical results show the feasibility of PIN cyber-attacks based on motion sensors, with no restrictions on the PIN

length and on the possible digit combinations. For example, 4-digit PINs are correctly recognized at the first attempt with an accuracy of 37%, and in five attempts with an accuracy of 63%.

E. Inference of Health Parameters and Body Features

Body-worn accelerometers provide valuable insights into a person's physical characteristics and health status. By analyzing accelerometer data from smartphones, researchers have been able to approximate users' body weight and height. There is a strong correlation between accelerometer-determined physical activity and obesity, making physical activity a recognized indicator of health. The amount of physical activity can reveal information about latent chronic diseases, mobility, cognitive function, and even the risk of mortality.

Accelerometer data allows for the derivation of various activity-related variables such as energy expenditure, activity type, and temporal activity patterns. These variables are increasingly used in health studies to remotely assess participants' physical activity levels. Sleep duration is another important factor in population health, and accelerometers in wearable devices have been utilized to evaluate sleep patterns, fragmentation, and efficiency. Actigraphy, which uses accelerometers, is considered an essential tool in sleep research and sleep medicine.

Specialized accelerometers have been employed to measure additional health parameters, including voice health, postural stability, and physiological sound. The versatility of accelerometers makes them valuable in monitoring various aspects of an individual's well-being and can aid in healthcare research and personalized healthcare management [5].

F. Inference of Demographics

Data from body-worn accelerometers can be used to estimate demographic variables such as age and gender. Differences in walking smoothness between adults and children can be detected through accelerometer readings. Gait features, including step length, velocity, and step timing variability, vary between younger and older subjects. Using smartphone accelerometer data, researchers have achieved a 92.5% success rate in predicting the age interval of test subjects based on their smartphone holding and touching behaviors.

Gender-specific movement patterns have also been identified using accelerometer data. Hip movements, gait features, and physical activity patterns derived from accelerometers can be used to estimate the sex of individuals. Notably, accelerometer-based gender recognition can work independently of a person's weight and height. Additionally, acoustic vibrations captured through a smartphone accelerometer can be used to classify speakers as male or female with high accuracy [5] .

G. Mood and Emotion Recognition

Physical activity, as measured by body-worn accelerometers, has been linked to human emotionsand depressive moods. Researchers have used accelerometer data from smart wristbands to recognize emotional states, such as happiness, neutrality, and anger, with fair accuracy. Accelerometers in smartphones have been employed to detect stress levels and arousal in users. Additionally, there is a positive association between accelerometer-derived speech activity and mood changes [5].

H. Inference of Personality Traits

Methods have been developed to infer preferences and personality traits based on body gestures and motion patterns captured by accelerometers. Wearable accelerometers were used to estimate the motivations, interests, and group affiliations of study participants during social interactions, relying on their movements, body postures, and gesturing patterns.

Furthermore, a person's level of physical activity, which can be measured using body-worn accelerometers, has been found to correlate with specific personality traits. Studies have shown that conscientiousness, neuroticism, openness, and extraversion are associated with different levels of physical activity. For example, it has been found that agreeableness, conscientiousness, and extraversion were positively correlated with higher step counts and physical activity variables, while neuroticism showed a negative association. Moreover, it has been discovered that neuroticism and the functioning of the behavioral inhibition system were related to physical activity measures derived from accelerometer data in female college students [5].

I. Discussion

The previous sections have highlighted the potential privacy invasions that can arise from accelerometers in mobile devices. A visual overview of all privacy threats is provided in Fig. 4. Despite being considered non-intrusive, accelerometer data can reveal sensitive information about a user's location, health, body features, age, gender, emotions, and personality traits. It can even be used for biometric identification and reconstructing text sequences including passwords or PINs. However, it's important to acknowledge the limitations of many experimental studies in this field. Most approaches have been tested in controlled laboratory settings, and real-life conditions may result in reduced accuracy. Some methods also rely on prior knowledge about the user or context, which may not always be available.

While the limitations of current research should be recognized, it is reasonable to assume that parties with regular access to accelerometer data, such as device manufacturers, service providers, and app developers, may possess more extensive training data, technical expertise, and financial resources than academic researchers. Moreover, these potential adversaries may have access to data from other sensors and auxiliary sources, further enhancing their ability to draw sensitive inferences. Therefore, this paper represents only an initial exploration of the topic, and the real-world privacy implications may be more significant than what has been identified so far.

Even if only one of the identified threats materializes, it can have serious implications for user privacy. As sensor

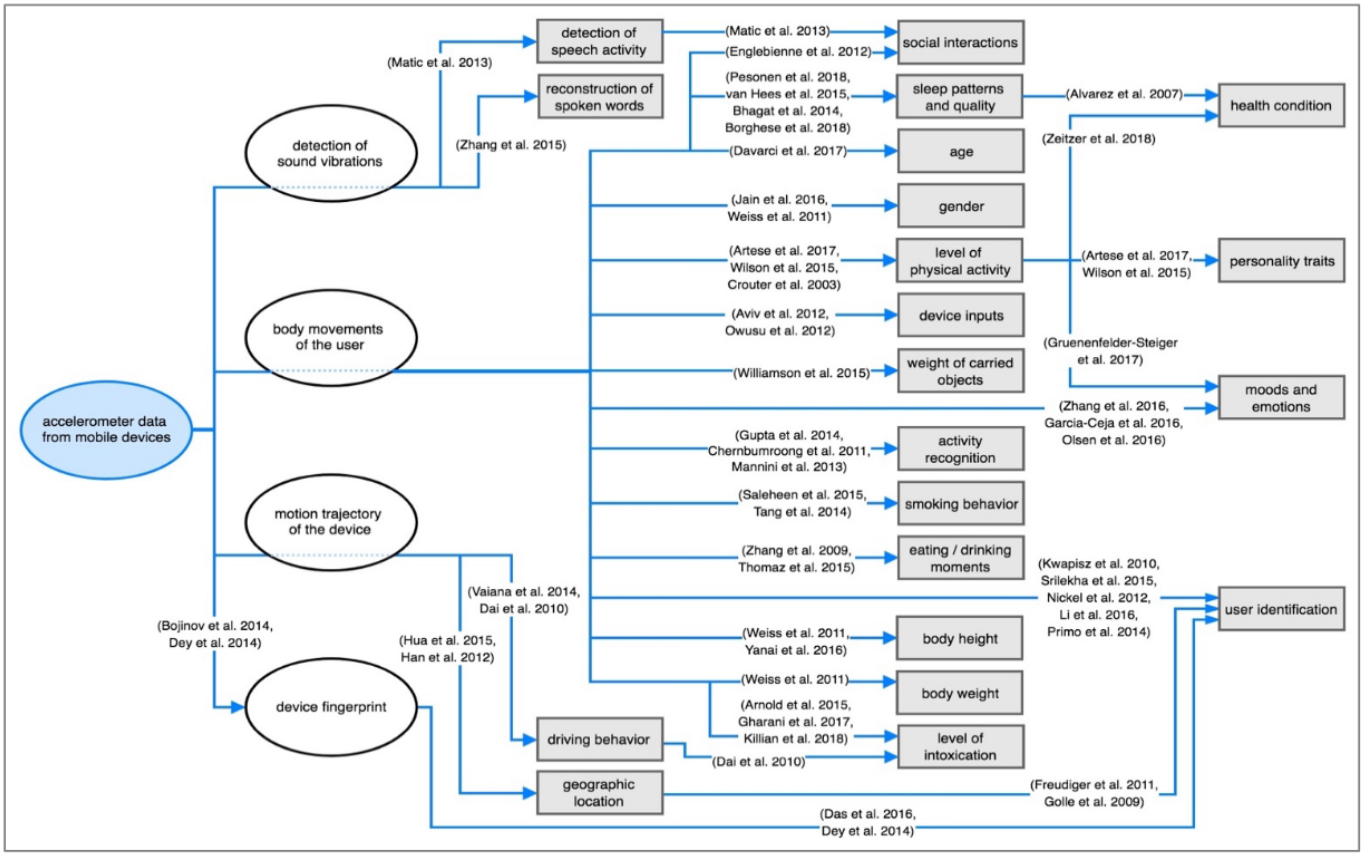

Fig. 4. Overview of sensitive inferences that can be drawn from accelerometer data (according to the referenced studies). The Fig. is taken from [5].

technologies continue to improve in terms of cost, size, and accuracy, and machine learning methods advance, along with the widespread adoption of accelerometer-equipped devices, the risks are likely to escalate. It is crucial to reconsider the privacy implications of accelerometers and implement corresponding technical and legal protection measures. The sensitivity of sensor data should be assessed based on all plausible inferences that can be drawn from it, rather than relying solely on the sensor's official purpose. Further research is needed to explore the privacy intrusion potential of accelerometers and other seemingly benign sensors, taking into account state-of-the-art data mining techniques. Given the difficulty of determining the limits of advancing inference methods, it is advisable to treat most sensors in mobile devices as highly sensitive by default [5].

V. Inferring Typed Words

This section examines the research conducted by Wang et al. [3] on the potential information leakage from motion sensors in smartwatches, specifically the accelerometer and gyroscope data, in relation to the user's typing activities on a laptop keyboard. The paper highlights the significant ramifications of such data leakage, as smartwatches can be camouflaged as activity trackers, thereby compromising the privacy of a user's emails, search queries, and other documents typed on the keyboard. Given the importance of addressing privacy and security risks associated with accelerometer data, this paper has been selected for further investigation and analysis.

Unlike keystroke loggers that need to find loopholes in the operating system, the activity tracker malware can obtain the user's permission and easily launch a side channel attack. They have been shown that when a user types a word W, it is possible to shortlist a median of 24 words, such that W is in this shortlist in English language under the condition that the watch is worn only on the left hand. When the word is longer than 6 characters, the median shortlist drops to 10. Additionally, other "leaks" are mentioned that can further reduce the shortlist and offer starting points for future work [3].

A. Constraints

The absence of data from the right hand is a unique constraint, and so it needs to infer which finger executed the key-press. For a given position of the wrist watch, it is not obvious which one of the 3 or 4 different keys could have been pressed, which could be further interspersed by unknown number of keys pressed by the right hand. Moreover, users write different with dexterity, e.g. some use their little finger far less efficiently while others use specific fingers when it comes to digits or corner keys [3].

B. Prerequisites

Two authors put on Samsung Gear Live smart watches and typed 500 words each wearing the smart watch on their left wrist. The accelerometer and gyroscope data is used as training data, and processed through a sequence of steps, including key-press detection, hand-motion tracking, character point cloud computation, and Bayesian modeling and inference. The test data was collect by 8 different volunteers who were asked to type 300 different English words from a dictionary. The smart-watch sensor data from the volunteers was used to create a short-lists K words, ranked in the decreasing order of probability (i.e., the first ranked word is considered the most probable guess) [3].

C. Problem of Capturing Smartwatch Data

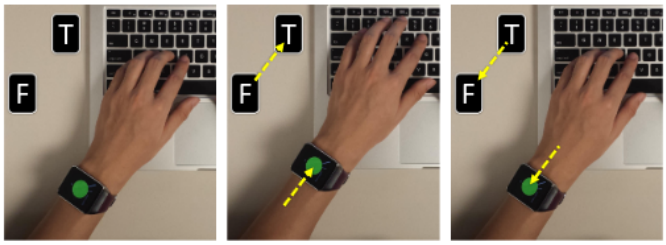

Fig. 5. 3 video frames show the process of typing "T" from "F". The figure is taken from [3].

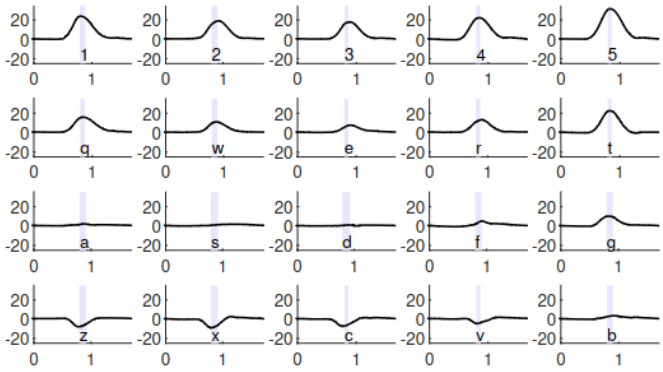

Fig. 6. The watch X axis displacements while a human types 20 characters. In the figures, X axis is time in seconds and Y axis is watch X axis displacement in millimeter. The gray bar shows the keystroke press and release time interval. The figure is taken from [3].

Two of the authors wore a smartwatch and recorded the accelerometer and gyroscope data as they typed each character one by one. The positive X axis of the watch is parallel to the arm and pointed towards the fingers, the positive Y axis is perpendicular and upward, and the positive Z axis pointed upwards from the plane of the arm.

In order to collect the ground truth data, a phone camera was placed right on top of the keyboard and recorded video at 30 fps and smart watch movements were captured by computer vision techniques. Figure 5 shows an example sequence of

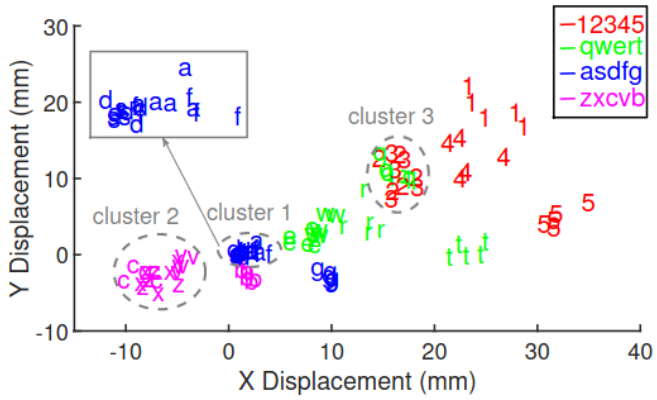

Fig. 7. Watch 2D displacements while a human types 20 characters with her left hand. Each character is typed repeatedly 5 times. (0,0) is the initial location when left hand fingers are placed on home position ("asdf"). Note that X and Y axes in the graph are in the watch's coordinate system. The figure is taken from [3].

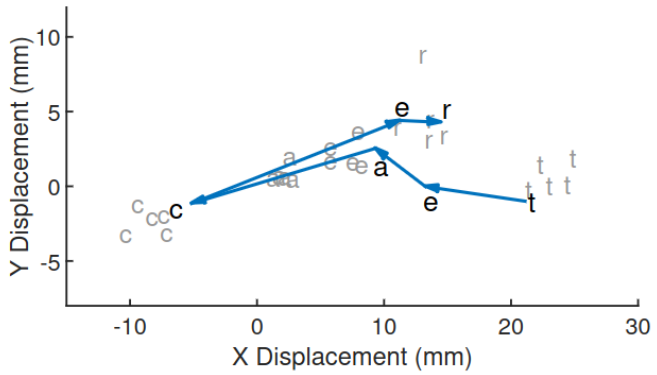

Fig. 8. Comparison of typing "teacher" continuously (in black) against each character separately (in gray). Note that the positions of "e", "a" and "c" are away from their original points due to sequential typing. Also, "h" is not captured due to right hand typing. The figure is taken from [3].

video frames capturing the process of typing the character "T". The left hand starts from a home position (i.e., the key "F"), moves along the $+X$ direction to press "T", hits the key, and returns back to the home position. The yellow arrow on the arm shows the displacement of the green marker on the watch. In Figure 6, motion data from 20 different characters located on the left side of the keyboard is plotted. Each graph shows the displacement of the watch computed from the accelerometer's X axis data, with the X axis representing time and the Y axis representing displacement. The accelerometer's Y and Z axes are not shown. The time of the key-press, obtained from the keyboard logger, is marked by a light gray vertical bar in each graph. It can be observed that the displacements align well with the layout of the keyboard. The first row (12345) generates the largest positive displacement, while the last row (zxcvb) produces negative displacement. As the left fingers are initially placed on the third row (asdf), nearly no displacement is detected for these characters. The authors suggest that although this is preliminary research,

these signals provide the first indication of information leakage through smartwatches.

In Figure 7, the watch displacement for the same 20 keys is shown in 2D space using the combined X and Y axes data from the accelerometer. Each color represents one row on the keyboard. Some keys, such as 1, t, r, 4, and 5, are quite isolated, while others overlap strongly, particularly "asdf," "zxcv," and "q23" exhibit the strongest overlaps. This is not surprising since the cluster "asdf" is an outcome of the fingers being on these keys in the home position, and the wrist hardly needs to move when typing these keys. Similarly, the fingers move uniformly downward for "zxcv," resulting in similarity between the keys. Lastly, the hand movement for "q" is similar to "2" and "3," even though they are not on the same row. This is because the little finger is shorter, and to type the character "q," it must move as much as the ring finger must move to type "2."

Decoding characters becomes more complex when the user types a word rather than just a single character. Figure 8 displays the sequence of hand displacements when the word "teacher" is typed. Obvious issues arise: The wrist motion for each character is no longer aligned with the earlier observations since the motion is relative to the previous position of the key. It can be observed that "e," "a," and "c" are all far away from their respective clusters detected earlier in Figure 7. Additionally, "h" (pressed by the right hand) was not recorded, and instead, a small random motion of the left hand during this time was captured. Finally, real-world environments do not have cameras, so the data is completely unlabeled. A wrong decision about any of the keys can derail all subsequent decisions. In conclusion, while sensor data from smartwatches can encode the human-typed information, decoding them reliably in real-world conditions presents non-trivial challenges.

D. Overievw of the MoLe Approach

Figure 9 illustrates the flow of operations in the end to end MoLe system. At the backend server, the attacker types each character on a computer keyboard multiple times and computes a character point cloud (CPC) similar to the one in Figure 7. The operation is performed offline, and is stored separately for use later.

The process of decoding the raw sensor data involves passing it through a module called "Keystroke Detection", which has two tasks. Firstly, it detects the timing of each keystroke by analyzing the Z axis of the sensor data, where a negative dip in the Z axis corresponds to a key press. Secondly, it computes the net 2D displacement of the watch by processing the signal through several steps, including gravity and mean removal, double integration, and Kalman Filtering. The output of this module is a set of tuples representing the estimated location of the watch at the time of each key press. These tuples form an unlabeled point cloud (UPC) on a 2D plane. The UPC is then passed to the "Cloud Fitting" module, which assigns approximate labels to the points by scaling and rotating the convex hull of a previously computed character point cloud (CPC) to best fit the convex hull of the UPC. This rotated and scaled CPC serves as the reference template for decoding the unlabeled points in the UPC.

The **Cloud Fitting** module receives the UPC and its primary responsibility is to provide estimated labels for the points in the UPC. To achieve this, the module uses the previously computed CPC and adjusts the scaling and rotation of the CPC's convex hull to match the convex hull of the UPC as closely as possible. The resulting rotated and scaled CPC serves as a reference template for identifying the characters corresponding to the unlabeled points in the UPC

The **Bayesian Inference** module takes in three inputs: (1) the template output from Cloud Fitting, (2) the unlabeled points from the UPC, and (3) a dictionary W of valid English words. For each valid word w_i in the dictionary, the module computes the a posteriori probability that the unlabeled points form w_i. This is done by computing the probability that each unlabeled point corresponds to a specific character in the word. The product of these probabilities is the final probability that the unlabeled points form the word w_i. The module computes this probability for each word w_i and outputs a ranked list of $<$word, probability$>$ tuples as a guess of the user-typed word. If the input is a password, the attacker can try out all the guesses above some probability threshold. If the input is an email or a search query, the attacker could manually try to decode the text from the possible sets of words. Even though MoLe does not offer a single suggestion, the probability estimate associated with each guess dramatically reduces the search space for the attacker.

1) Assumptions: This paper defines the following conditions for their experiment:

- Volunteers type one word at a time (as opposed to free-flowing sentences).
- Only valid English words are allowed. Passwords that contain interspersed digits, or non-English character-sequences, are not decodable as of now.
- The same Samsung smart watch model was used for both the attacker and the user.
- They assumed the user is seasoned in typing in that he/she roughly uses the appropriate fingers – novice typists who do not abide by basic typing rules may not be subject to our proposed attacks.

E. Keystroke Detector

This module is tasked with computing the timing and position of each key press present in the sensor signals. The position is represented by a 2D vector with the origin at the "F" key on the keyboard. Collecting all of the key press positions will result in the point cloud as discussed previously.

1) Key-press Timing: The method to detect key presses is based on the idea that when a finger is pressed on a key, the wrist undergoes a partial dipping motion which can be observed in the Z-axis of the watch. The Z-axis motion when a user types the word "administrative" is shown in Figure 10. Although actual key presses generally produce prominent peaks, false positives and false negatives can occur.

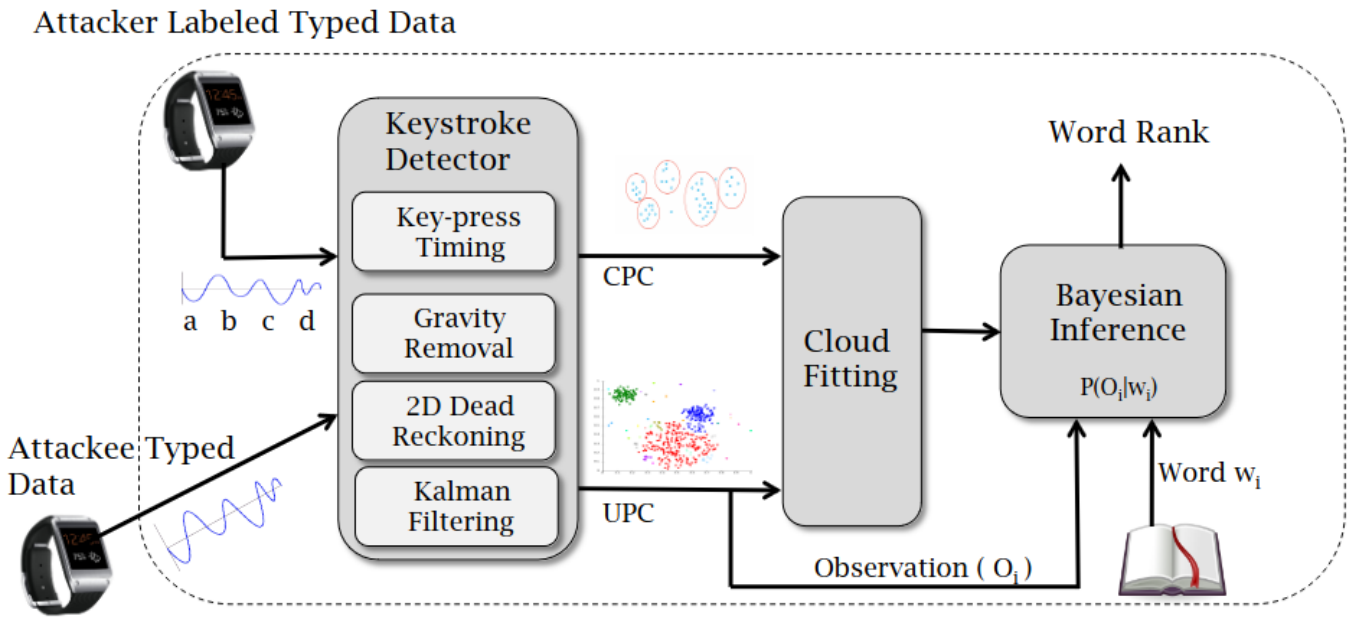

Fig. 9. The typed data from users are pre-processed through gravity removal and timing analysis blocks, super-imposed on the refitted typing templates, and passed through a Bayesian inference model that leverages the patterns and structures in English words to ultimately decode the typed words. Note, training is only required from the attacker's end; no training needed for the user. The figure is taken from [3].

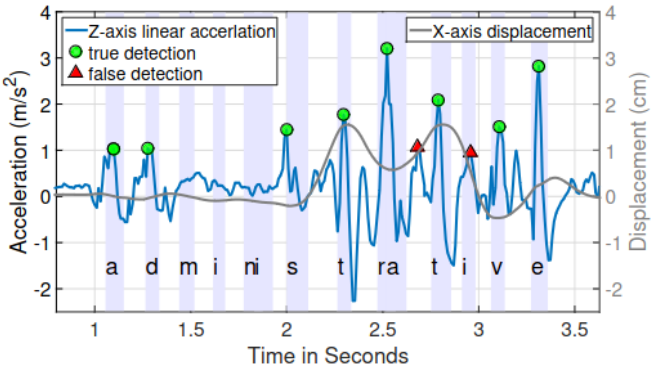

Fig. 10. A simple peak detection scheme to detect keystrokes. The left Y -axis represents acceleration and the right Y -axis indicates displacement. Note that, for "a","d","s" keystrokes, lower Z-axis acceleration is generated because of left hand's initial position. At time 2.7 and 3 seconds, there are two false detections due to the left hand moving from "a" to "t" and from "t" to "v". The figure is taken from [3].

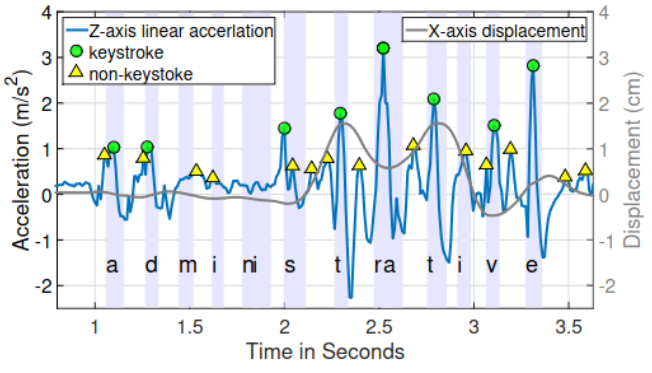

Fig. 11. Bagged decision classification results: A peak detection tool with low thresholds is first applied to the Z-axis acceleration data and marks potential keystrokes (both Yellow tri-angles and green circles). The classifier then identifies whether the peaks are keystrokes or not. Note that for the first "a" and "d", since two peaks are too close, the classifier would identify only one peak with highest Z-axis acceleration within a time window. The figure is taken from [3].

False positives are mainly observed during transitions between keys, where the hand moves up slightly before making the movement, which can be reflected in the Z-axis motion. False negatives are usually due to subtle Z-axis motion for keys like "asdf" that may not be detected. Ground truth is used to observe these false positives and false negatives.

To address false positives and false negatives when detecting keystrokes, bagged decision trees are used as an ensemble classifier that trains multiple decision trees by selecting different subsets of features and training examples. This approach improves the stability and accuracy by letting each subtree learn on the attacker's labeled data, apply the learning to the unlabeled data, and then compute the final results via voting. To obtain labeled data, a simple threshold-based peak detection method is applied on the Z-axis acceleration, and true/false detection is labeled on the attacker's template. The peak detection threshold is set to be low so as not to miss true keystrokes. Features are extracted within a time window around the labels, and the classifier is trained using these features.

The feature set includes various metrics such as width,

height, and prominence of the Z-axis peak, mean, variance, max, min, skewness, and kurtosis for each of the 3-axis displacement, velocity, acceleration, and gyroscope rotation, the magnitude of acceleration/gyroscope, and the correlation of each pair between acceleration and gyroscope vectors. When the attacker's sensor data arrives, the same peak detection scheme is applied, and many candidate keystrokes and their features are obtained. The classifier identifies the validity of the keystroke and selects the maximum value of Z-axis acceleration to denote the timing of the key-press. An example of the classification result for the word "administrative" is shown in Figure 11.

2) Key-Press Location Estimation: The primary challenge is accurately tracking hand motion during key transitions to determine key-press locations. It requires high accuracy, as errors can cascade. The left index finger's periodic return to the home key "F" allows recalibration and improves tracking accuracy. Initially, Android API's linear acceleration data was used but it was unsatisfactory. So the authors developed a tailored tracking approach by themselves. Steps include finding gravity to establish an absolute coordinate system, estimating and removing gravity using gyroscope data, estimating coordinates and calculating projected acceleration, and calibrating speed and displacement through mean removal. A Kalman smoothing was also employed for displacement estimation stability.

3) Point Cloud Fitting: MoLe generates an unlabeled point cloud (UPC) based on the estimated displacements for each key pressed by the attacker. To assign approximate labels to the points in the UPC, MoLe fits the attacker's character point cloud (CPC) to the UPC. The fitting process involves computing convex hulls for both the CPC and the UPC. MoLe computes two convex hulls for the CPC, one for positive X displacements (H_{pos}^{CPC}) and another for negative X displacements (H_{neg}^{CPC}). Similarly, two convex hulls are computed for the UPC (H_{pos}^{CPC} and H_{neg}^{UPC}). The fitting is performed by rotating and scaling the CPC's convex hulls to align with the UPC's convex hulls. The fitting metric is based on the degree of overlap between the two convex hulls, specifically the ratio of their intersection and union. Figure 12 shows an example of point cloud fitting.

In cases where the attacker generates multiple CPCs, MoLe performs the fitting process for each CPC and selects the one that maximizes the intersection/union ratio. Once the CPC is rotated and scaled, it is superimposed on the UPC, creating a framework for estimating labels for each point in the UPC.

4) Bayesian Inference: After detecting keystrokes and fitting the point cloud, MoLe aims to infer the characters typed by the right hand, filling in the missing information. One approach is to calculate the posterior probability of each word in the English dictionary given the motion inferences from the left hand. The Bayesian inference step involves calculating the posterior probability of each word in the English dictionary given the observed motion data from the left hand. By applying Bayes' theorem, the posterior probability is obtained:

$$P(W|O) = P(O|W) * P(W)/P(O)$$

Where:

- W represents a candidate word from the dictionary and O represents the observed motion data.
- $P(W|O)$ is the posterior probability of the word given the observed motion data.
- $P(O|W)$ is the likelihood function that estimates the probability of the word W based on the observed motion data.
- $P(W)$ is the prior probability that captures the occurrence frequency of the word.
- $P(O)$ is the probability of the observation, which is the same for all possible words and can be ignored in the calculation.

The goal is to find the word with the highest likelihood $P(O|W)$ given the observed data. Various refinements can be applied to improve the likelihood function and posterior probability estimation.

Using the Number of Keystrokes. One approach is to consider the number of detected keystrokes as observations to match the word. The number of keys typed by the left hand is used to evaluate the likelihood of each word. Each word is compared based on the detected number of keys. For example, when two keys are detected, the word "the" is more likely to produce a higher likelihood than the word "teacher" because the number of peaks generated while typing "the" is much closer to 2 than "teacher". Now, for each detected keystroke, it is desired to match them with the characters in the word. Since the keystroke could be caused by any character in the word, all possible assignments need to be considered. The posterior probability can be calculated as follows:

$$P(O|W) = P(N|W) = \sum_{(\alpha_1,...,\alpha_N)} P((c_{\alpha_1},...,c_{\alpha_N})|W)$$

where N is the number of keystrokes and $(\alpha_1, ..., \alpha_N)$ represents one possible N -element combination from 1, 2, ..., L and L is the word length. The summation adds up all possible combinations. c_i is the i-th character in W; $P((c_{\alpha_1}, ..., c_{\alpha_N})|W)$ is probability that N peaks are generated by $P((c_{\alpha_1}, ..., c_{\alpha_N})|W)$.

Thus, by iterating over all words in the dictionary, we can obtain the likelihood for each word.

Consecutive Characters. Additionally, for cases where two consecutive characters are detected as a single keystroke due to close succession, treating them as one key-press can be appropriate as for "er", "sa" or "re". These keys are adjacent on the keyboard and the watch dips in so close succession that they are not separable.

Adding Watch Displacement. The actual watch displacement is now ready to be utilized by MoLe. Assuming that fingers are positioned over the home position ("F" and "J"), the key-press location estimation module calculates the location of each key press (Figure 7). However, considering that the Conditional Positional Constraints (CPC) has been adjusted to the user's User Positional Constraints (UPC), it is now feasible to improve word prediction by incorporating displacement.

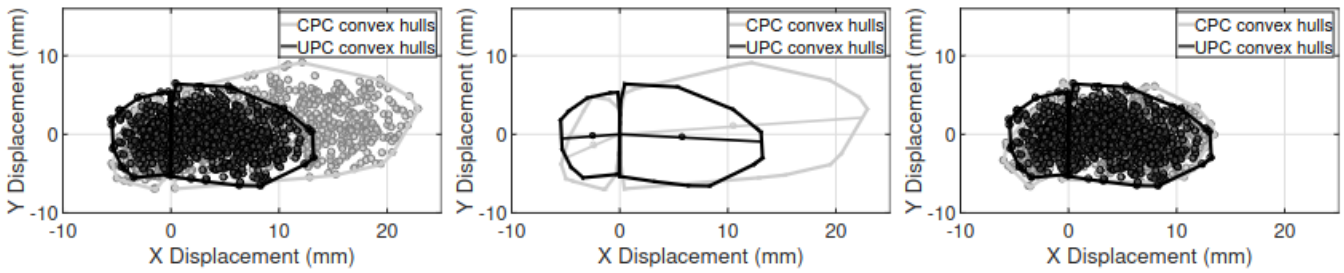

Figure 13: Point Cloud Fitting. Black points are the CPC attacker template and gray points are UPC from attackee. (a) Finding each convex hull (b) Calculate the centroids and perform rotate and scale. (c) Point cloud fitting result.

Fig. 12. Point Cloud Fitting. Black points are the CPC attacker template and gray points are UPC from attackee. (a) Finding each convex hull (b) Calculate the centroids and perform rotate and scale. (c) Point cloud fitting result. The figure is taken from [3].

Since MoLe models each character's location as a Gaussian distribution and assuming the distribution of displacement d_i only depends on current character c_{α_i}, the posterior probability can be reformulated as follows:

$$P(O|W) = P(N \cap d_i, i = 1, 2, ..., N|W)$$

$$= \sum_{(\alpha_1,...,\alpha_N)} P((c_{\alpha_1}, ..., c_{\alpha_N})|W) \prod_{i=1}^{N} p(d_i|c_{\alpha_i})$$

Character Transitions. It was previously assumed that each character displacement is independent. However, when typing a word, sequential movements occur, and the current displacement is influenced by the location of the preceding character. It is evident that the displacement distributions differ. In the case of "ra", the displacement of typing "a" is shifted towards the position of the character "r" because the little finger types "a" immediately after "r" before returning to the home position. Conversely, in the case of "va", the displacement of "a" is closer to the position of "v" due to the same reason. Thus, the likelihood function extends to the following:

$$P(O|W) = \sum_{(\alpha_1,...,\alpha_N)} P((c_{\alpha_1}, ..., c_{\alpha_N})|W) \prod_{i=1}^{N} p(d_i|c_{\alpha_i}, c_{\alpha_{i-1}})$$

Keystroke Interval. The timing of key-presses on the left hand can encode information about missing keys. The probability of having N right hand characters between consecutive keystrokes can be determined based on the detected time interval. Accurate estimations of N can contribute to the correct predictions. For instance, when typing the word "thanks" (with characters typed on the left hand underlined), the observed interval between "t" and "a" is expected to be shorter compared to the interval between "a" and "s".

The time interval generally increases as the number of keystrokes increases. However, there is significant variance observed. For instance, although the segment "-b i l i t-" and "-t i o n a-" both contain three right hand characters in the middle, the average interval of "-t i o n a-" is shorter than "-b i l i t-" due to factors like hand geometry and typing familiarity.

The observation can be written into:

$$P(O|W) = P(N \cap d_i, i = 1, 2, ..., N \cap_{t_j}, j = 1, 2, ..., N-1|W)$$

$$= \sum_{(\alpha_1,...,\alpha_N)} P((c_{\alpha_1}, ..., c_{\alpha_N})|W) \prod_{i=1}^{N} p(d_i|c_{\alpha_i}) p((t_1, ..., t_{N-1})$$

$$|(c_{\alpha_1}, ..., c_{\alpha_N}), (d_1, ..., d_N), W)$$

F. Evaluation

MoLe was implemented on the Galaxy Gear Live smartwatch. The MoLe client on the watch continuously logs accelerometer and gyroscope readings at 200Hz, along with timestamps. The sensor data is stored locally during data collection and later transferred to a backend MATLAB server for analysis.

For evaluation, eight volunteers were recruited through advertising on the university campus and familiar with English typing. Each was asked to type 300 English words randomly selected from the 5000 most frequently used words. The word length ranged from 1 to 14, and each length was equally distributed. A total of 2400 words were tested across all users.

During the experiment, subjects were seated at a desk in front of a Lenovo laptop. A word appeared one at a time on the laptop screen, and the subjects were instructed to type the same word in a text box on the screen. If any characters were mistyped, the data was discarded, and the subject was asked to re-enter the word. Subjects were instructed to initialize their hand position on the "F" and "J" keys between each word recording. The laptop recorded the timing of the keystrokes, which served as the ground truth.

To collect offline training data, two of the authors acted as attackers and followed the same procedure, but with the top 500 longest words in the dictionary. The use of long words helped capture a diverse range of typing patterns. The data

collection for the offline training was done using a Lenovo ThinkPad equipped with a regular full-sized keyboard.

For full ground truth recording, an Android Samsung Galaxy S4 phone was mounted on top of the keyboard, and the front camera was used to capture video of hand movement. The camera calibration toolbox in MATLAB was used to calibrate the camera pixel and measure the watch distance and location from each frame.

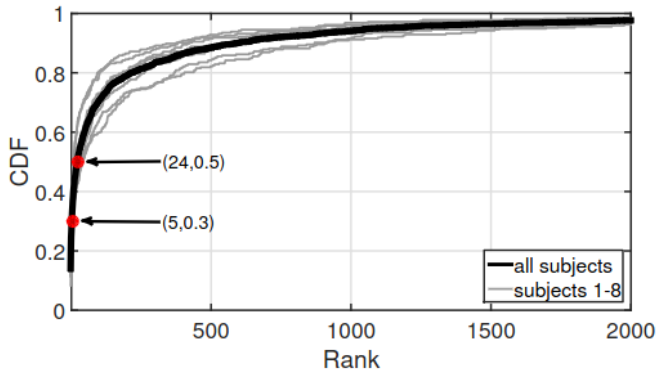

Fig. 13. CDF of rank computed for each of the 2400 words typed by 8 subjects. The figure is taken from [3].

Figure 13 illustrates the cumulative distribution function (CDF) of rank, based on the analysis of 2400 words typed by the subjects in the experiments. The black line represents the average performance across all 8 subjects, while the gray lines represent MoLe's performance for each individual subject. The results indicate that the median rank of a word is 24, i.e. there is a 50% chance that MoLe can narrow down the typed word to 24 possibilities. Additionally, at the 30th percentile, the rank is 5, indicating a 30% chance of narrowing down the possibilities to just 5 words. This reduction in the search space, considering a total of 5000 possible words, is significant and increases susceptibility to brute-force attacks.

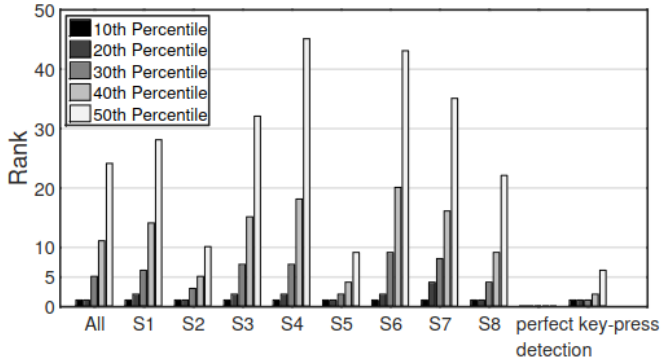

Fig. 14. Rank of average, across users and with perfect key-press detection. The figure is taken from [3].

Figure 14 displays the ranks of typed words for each test subject. It is observed that subjects S2, S5, and S8 generally have higher ranks, indicating that MoLe's guesses are more accurate for these individuals. Upon analyzing the video and sensor traces, it is discovered that these subjects exhibit lower variance in their hand movements, likely due to adhering to the prescribed typing guidelines. Furthermore, a test is conducted with perfect key-press detection, utilizing the actual number and timing of keystrokes from the left hand, obtained from the ground truth timing information. Surprisingly, the 30th percentile drops to 1, indicating that MoLe can precisely guess the word, and the 50th percentile drops to 6. This outcome highlights that further enhancements in key-press detection are crucial for the improvement of MoLe's performance.

When the left character count ranges from 2 to 4, the performance of MoLe tends to degrade. This is because there are many words that share the same 2 to 4 left-hand characters. The similarity in the initial characters makes it more difficult for MoLe to accurately narrow down the possibilities and correctly identify the typed word. The rank generally decreases as the word length exceeds 6. This is primarily due to two reasons. Firstly, longer words tend to have a greater number of keystrokes, which increases the chances of accurate detection. Secondly, the number of words with longer lengths decreases, reducing the possibilities for confusion. Conversely, words with lengths ranging from 4 to 7 have fewer keystrokes, making them more challenging to detect. Additionally, there are a larger number of words with these lengths, further adding to the difficulty of detection.

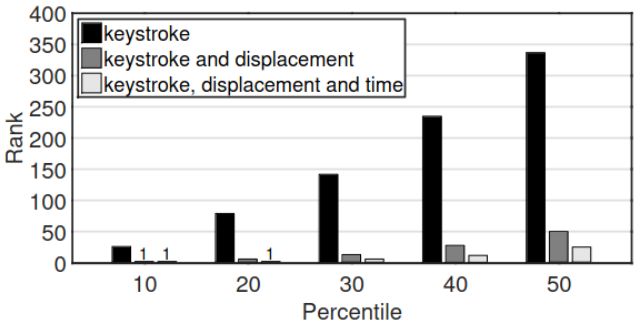

Fig. 15. Contribution of different opportunities towards the overall performance of MoLe. The figure is taken from [3].

Figure 15 presents the breakdown of contributions from each of the factors. When utilizing only the number of detected keystrokes (refinements 1 and 2), MoLe demonstrates poor performance on the dataset, with a rank of 340. However, when incorporating displacement information (refinements 3 and 4), MoLe significantly improves its rank, reaching 49th percentile. Finally, with the inclusion of time intervals (refinement 5), the median rank further improves from 49 to 24.

Evidently, MoLe's ability to guess degrades drastically with lower sampling rates – median ranking falls as 64, 141 and 1218. The authors state that the system performance is close between two keyboards, even though the attackers used the laptop keyboard for training the system.

Table 16 displays MoLe's end-to-end prediction results for each word in an actual sentence entered by subject S5. The

Rank	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8
1.	motor	pistol	profound	technology	angel	those	that	disappear
2.	monitor	list	journalism	remaining	spray	today	tight	discourse
3.	them	but	originally	telephone	super	third	tightly	secondary
4.	the	lost	original	meanwhile	fire	through	thirty	adviser
5.	then	most	profile	headline	shore	towel	truth	discover

Fig. 16. The table shows an example of predicted words according to the rank. Readers can reconstruct the sentence: "The most profound technologies are those that disappear". The table is taken from [3].

table lists the Top-5 guesses for each word, with the most likely guess at the top. The words in each column exhibit similarity in their character sequences. By examining the table, readers can reconstruct the sentence: "The most profound technologies are those that disappear".

1) Limitations: MoLe is not yet a real-world attack since it is not able to infer non-valid English words, such as passwords, inability to parse sentences due to difficulties in detecting the "space bar". Additionally, training and test data contain no mistakes such as mistyping and pressing delete key. Moreover, subjects were instructed to follow the typing guideline such as returning to "F"-Key. However, the authors state that although no tests were made with other wearable devices such as Fitbits, they believe with some customization, the attacks can be launched on those platforms as well.

VI. Conclusion

In conclusion, wearable devices have gained significant popularity due to their convenience and versatility, enabling users to engage in various activities such as making payments, monitoring health, and receiving notifications. However, the widespread adoption of wearables also introduces security and privacy challenges that must be addressed. This report has provided an overview of the security and privacy considerations associated with IoT wearables, highlighting the critical importance of implementing robust security measures and privacy safeguards.

Specifically, the report has focused on the potential threats posed by accelerometer and gyroscope data collected by wearables. These motion sensors, while providing valuable functionality, also pose risks to user security and privacy. Privacy vulnerabilities associated with mobile device sensors, specifically motion sensors, have been identified, giving rise to concerns regarding location tracking, audio recording, and keystroke inference attacks. Furthermore, the report has delved into the findings of a research paper that examines the vulnerabilities associated with analyzing motion sensor data to decipher keyboard inputs. In this paper, reasonable predictions can be made about the words being typed by analyzing the accelerometer and gyroscope signals, capturing wrist micromotions, and incorporating the structure of valid English words. This type of attack can significantly compromise the privacy of individuals, especially considering the increasing popularity of smartwatch app stores. The findings underscore the need for effective security measures in wearable technologies. It is crucial to mitigate the identified security and privacy threats to ensure the trust and confidence of users. By addressing these vulnerabilities and implementing robust security protocols, wearable device manufacturers and service providers can enhance the overall security and privacy posture of these devices, promoting their safe and reliable use in various domains. According to [5], it is imperative for future research to prioritize the development of wearable devices that are both energy-efficient and secure, as well as user-friendly privacy controls and encryption techniques to ensure the safeguarding of user data. Additionally, the establishment of standardized protocols and regulatory frameworks is necessary to promote responsible and ethical usage of sensor data while upholding user privacy. By addressing these challenges and vulnerabilities, the capabilities of wearable devices and mobile sensors can be fully harnessed while simultaneously safeguarding user privacy within the ever-evolving landscape of technology.

References

[1] Seneviratne, Suranga et al. "A Survey of Wearable Devices and Challenges." IEEE Communications Surveys & Tutorials 19 (2017): 2573-2620.

[2] Delgado-Santos, Paula et al. "A Survey of Privacy Vulnerabilities of Mobile Device Sensors." ACM Computing Surveys (CSUR) 54 (2022): 1 - 30.

[3] Wang, He et al. "MoLe: Motion Leaks through Smartwatch Sensors." Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (2015): n. pag.

[4] Liu, Xiangyu et al. "When Good Becomes Evil: Keystroke Inference with Smartwatch." Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (2015): n. pag.

[5] Kröger, Jacob Leon et al. "Privacy implications of accelerometer data: a review of possible inferences." Proceedings of the 3rd International Conference on Cryptography, Security and Privacy (2019): n. pag.

[6] R. Goyal, N. Dragoni, and A. Spognardi, "Mind the tracker You wear: A security analysis of wearable health trackers," in Proceedings of the 31st Annual ACM Symposium on Applied Computing, ser. SAC '16. New York, NY, USA: ACM, 2016, pp. 131–136.

[7] https://openeffect.ca/reports/Every_Step_You_Fake.pdf, 2016.

[8] Sandra Gabriele and Sonia Chiasson. 2020. Understanding Fitness Tracker Users' Security and Privacy Knowledge, Attitudes and Behaviours. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20). Association for Computing Machinery, New York, NY, USA, 1–12.

[9] Lev Velykoivanenko, Kavous Salehzadeh Niksirat, Noé Zufferey, Mathias Humbert, Kévin Huguenin, and Mauro Cherubini. 2022. Are Those Steps Worth Your Privacy? Fitness-Tracker Users' Perceptions of Privacy and Utility. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 5, 4, Article 181 (Dec 2021), 41 pages.

[10] Crouter, S. et al. 2003. Validity of 10 Electronic Pedometers for Measuring Steps, Distance, and Energy Cost. Med. Sci. Sport Exer. 35, (2003), 1455–60

[11] Migueles, J. et al. 2017. Accelerometer Data Collection and Processing Criteria to Assess Physical Activity and Other Outcomes: A Systematic Review and Practical Considerations. Sports Medicine. 47, (2017).

[12] Yang, C.-C. and Hsu, Y.-L. 2010. A Review of Accelerometry-Based Wearable Motion Detectors for Physical Activity Monitoring. Sensors. 10, 8 (Aug. 2010), 7772–7788. DOI:https://doi.org/10.3390/s100807772.

[13] Sarkisyan, Allen, Ryan Debbiny and Ani Nahapetian. "WristSnoop: Smartphone PINs prediction using smartwatch motion sensors." 2015 IEEE International Workshop on Information Forensics and Security (WIFS) (2015): 1-6.

[14] Nerini, Matteo et al. "Machine Learning for PIN Side-Channel Attacks Based on Smartphone Motion Sensors." IEEE Access 11 (2023): 23008-23018.

[15] Das, Aveek K. et al. "Uncovering Privacy Leakage in BLE Network Traffic of Wearable Fitness Trackers." Proceedings of the 17th International Workshop on Mobile Computing Systems and Applications (2016): n. pag.