Intercepting Bluetooth Traffic from Wearable Health Devices

Qi Liu
University of Virginia
ql8va@virginia.edu

Yizhe Zhang
University of Virginia
yz6me@virginia.edu

Yixin Sun
University of Virginia
ys3kz@virginia.edu

2024 IEEE Security and Privacy Workshops (SPW) | 979-8-3503-5487-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/SPW63631.2024.00032

***Abstract*—Smart wearable devices are increasingly used to track health conditions and monitor health-related activities, such as blood pressure monitors, oximeters, and smartwatches. Such smart wearable devices often rely on Bluetooth Low Energy (BLE) to send health measurement data to the smartphone, which may then use Wi-Fi to sync up data to the cloud. Several recent works have explored passive attacks on the BLE or Wi-Fi or WAN traffic to infer user activities through the packet metadata. In our work, we take a first step towards investigating the effectiveness of active attacks that intercept the Bluetooth connection between the device and phone, enabling the adversary to extract user health data from encrypted Bluetooth packets which cannot be observed by passive attackers. We find that several popular wearable health devices are vulnerable to the attacks. The reason is rooted in the lack of security mechanisms adopted by these devices in their BLE implementations. Our work highlights the risks posed by Bluetooth traffic from wearable health devices and motivates the need to adopt secure Bluetooth practices to better protect user privacy.**

1. Introduction

Wearable devices are gaining increasing popularity in healthcare [11]. Users can monitor their health conditions or keep track of their exercise activities with off-the-shelf wearable devices, such as blood pressure monitors, oximeters, stress level monitors, and fitness trackers. The wearable devices typically transmit the measured data to the users' smartphones via Bluetooth Low Energy (BLE) [5]. Such data contains sensitive information about the users' health or exercise patterns.

Many prior works reveal privacy issues concerning smart home devices by performing passive traffic analysis on the WiFi or WAN traffic, i.e., traffic between device to cloud and between phone to cloud, to infer user activities at home [20, 21, 36, 26]. However, such attack method may not be effective for wearable health devices, many of which solely rely on Bluetooth to transmit the data to the phone and do not communicate directly with the cloud. A few recent works [25, 22] investigate the privacy issues surrounding the Bluetooth traffic between wearable devices and the phone. However, they only consider fitness trackers and smartwatches (e.g., Apple Watch, Fitbit), and only consider passive attacks.

To gain more insight into the security and privacy risks to users, we explore active attacks on BLE traffic from the wearable health devices. Our preliminary study includes 3 popular fitness trackers/smartwatches and 10 other top-seller healthcare devices, such as blood pressure monitors, oximeters, and stress level monitors. Our adversary model is a local adversary within the Bluetooth communication range who attempts to intercept and learn sensitive health information from the users. Examples of such adversary include nosy neighbors in apartment buildings or nearby guests in hotel rooms, employers who collect data from employees and sell to insurance companies [10], and advertisers in densely populated areas who try to learn user activities and profile the users [8, 6, 9, 7].

More specifically, we perform *active* Man-in-the-Middle (MITM) attacks to intercept the Bluetooth connection between the device and the phone, which enables the adversary to decode communication data that is otherwise invisible to passive eavesdroppers. By posing as a "fake" middle entity, the MITM adversary can prevent the device and the phone from communicating with each other (i.e., Denial-of-Service attack) or observe measurement data inside the encrypted packets sent from the device to the phone (i.e., data interception). We found that five wearable health devices are vulnerable to the DoS attack, among which three are vulnerable to data interception. Note that, unlike the passive attacker who can only observe metadata, the MITM adversary can see inside the encrypted packets and observe the actual measurement data, e.g., the measured oxygen level of the user from the oximeter.

Our preliminary study highlights the security and privacy risks of user health data when using wearable devices. We discuss potential countermeasures against the active attacks, such as Bluetooth security mechanisms which can make it significantly more difficult for adversaries to eavesdrop on a Bluetooth connection. We hope to raise user awareness of security and privacy issues when using wearable health devices, and urge the device vendors, manufacturers, and the research community to adopt Bluetooth best practices and work towards protecting sensitive health information from users.

DOI 10.1109/SPW63631.2024.00032

2. Background

In this section, we first provide background information on Bluetooth communications. We then discuss existing passive attacks on IoT devices.

2.1. Bluetooth Low Energy (BLE) Basics

Bluetooth is a wireless communication protocol that transmits data within 2.4GHz over a short range (e.g., between 10 to 100 meters) [1]. Bluetooth was originally designed for transmitting time-sensitive and continuous data (e.g., headphone audio data). With the emergence of IoT devices, which are lightweight devices that transmit data over a short period of time, a low-throughput, low-power protocol is needed. Bluetooth Low Energy (BLE) or Bluetooth 4.0 is a low-energy variation of Bluetooth, which is suitable and widely used in IoT devices [2]. Similar to Bluetooth, BLE also operates on 2.4 GHz. However, unlike Bluetooth, BLE remains in sleep mode to preserve energy consumption unless a connection is initiated. There are two communicating entities in the BLE protocol, BLE central (e.g., smartphones) and BLE peripheral (e.g., wearable devices such as blood pressure monitors). During a BLE communication, there are three main steps:

- **Connection:** The central and peripheral devices must go through the connection stage before starting the communication. During connection, devices broadcast their advertising packets, signaling their presence to nearby devices. At the same time, they also scan and listen to advertisement channels to discover targeted devices. Essential information, like BLE peripheral name, MAC address, services, and properties, are collected by the central device during scanning. Then, the central device initiates a connection with the targeted peripheral.
- **Paring:** Paring is used for encryption and is optional. Paring requires the two BLE devices to agree on a set of security parameters to derive a master key, i.e., the Long Term Key (LTK). There are four pairing methods, which will be elaborated in Section 5.1.
- **Communication:** Two entities start to exchange data and communicate through the Attribute Profile (ATT) protocol. ATT protocol is a server/client protocol where the peripheral device serves as the server, and the central device serves as the client. The server transmits the data it contains through its attributes, whereas the client reads or controls the servers' data through write/read requests.

All the wearable devices in our dataset use BLE.

2.2. Passive Traffic Analysis Attacks

Traffic analysis attack infers information by observing the communication pattern of the encrypted traffic. This technique has been adopted in various attack settings, such as website fingerprinting [32, 27, 30], social network analysis [28], identifying network devices and activities [33, 24], etc. With the emergence of Internet of Things (IoT), a number of recent works have developed traffic analysis attacks in smart home networks [36, 20, 21, 26, 18]. These attacks focus on extracting packet metadata from Wi-Fi or WAN traffic, such as packet lengths and timings, to uncover user activities and device types.

BLE is one of the most popular communication protocols in IoT devices because of its low power consumption. However, the research on passive traffic analysis attacks targeting BLE traffic remains limited. Das *et al.* [25] studies the privacy leakage in Fitness Tracker, analyzing BLE traffic patterns to infer the user's ongoing activity (e.g., walking, sitting, idle). Utilizing a professional BLE sniffer (ComProb BPA 600), they collected BLE traffic data from six popular fitness trackers. Their analysis revealed a strong correlation between BLE traffic rate and the user's activity intensity (motion). To validate this correlation, the authors utilized a decision tree with a feature vector comprising payload data rate, number of empty packets, and number of start packets, achieving a classification accuracy of 97.6%. Furthermore, the study distinguished individuals by their unique gait patterns, enabling personal identification, as different individuals exhibit distinct gait patterns, resulting in identifiable BLE traffic. The authors demonstrated high identification accuracy (over 90%) among five individuals. In a separate study, Barman *et al.* [22] investigated the Bluetooth metadata (e.g., packet time and length) exchanged between a wearable device and a phone. The authors curated a dataset comprising seven wearable bands. Employing machine learning techniques, they successfully extracted device information, human actions, app usage, and user profiles from the traffic data.

2.3. Active Attacks

Our work focuses on the vulnerability of BLE devices to active attacks, which involve a malicious attempt by unauthorized entities to disrupt or manipulate the data flow within the system. These attacks involve direct interference with data transmission rather than passive observation.

One form of active attacks targeting BLE is the Denial-of-Service (DoS) attack, where the attacker aims to make the system resource unavailable to legitimate users. A notable example is the battery exhaustion attack [34, 31], where the attacker depletes devices' battery power by keeping the device perpetually awake. BLE devices are also susceptible to the Denial-of-SLeep (DoSL) attack [37] where the attacker establishes continuous connections to the device to prevent it from entering sleep mode, thus resulting in significant power drain. Another significant active attack is the Man-in-the-Middle (MITM) attack, where the attacker intercepts and possibly alters communication between two parties without their knowledge. While prior works [19, 35] focus on MITM attacks in Bluetooth classic domain, there is a noticeable gap in the existing literature regarding active attacks in Bluetooth Low-Energy domain, particularly concerning wearable health devices.

2.4. Threat Model

We consider a local adversary who is within the Bluetooth communication range. We assume that the BLE adversary does not compromise any device and does not have the keys to decrypt the established connection between the device and the phone. The adversary can use a sniffer to capture or intercept all nearby Bluetooth traffic within the communication range. Note that although the traditional Bluetooth range is up to 100 feet or 30 meters, the distance can go up to hundreds of feet with the help of the BLE signal amplifier or repeater [3, 14].

While the active sniffer does not compromise any devices and does not possess the keys to decrypt the connection between the device and the phone, the sniffer could pose itself as the "phone" and directly interact with the device. Consequently, the active sniffer can gain visibility into the packet content, and inject or alter traffic.

The goals of the active sniffer are: (i) perform Denial of Service attack to prevent the device from connecting to the real smartphone, and (ii) extract sensitive health data from the encrypted packets sent by the device.

3. Dataset

We describe our wearable devices and data collection methodologies in this section.

3.1. Device Selection

We choose a diverse set of health-related wearable devices for our study, such as blood pressure monitor, oximeter, body scale, heart rate monitor, and smartwatches. The specific devices are selected based on the following criteria: (i) They are popular on Amazon [15] and are manufactured by popular vendors, e.g., Withings and Fitbit; (ii) They have good ratings on Amazon, i.e., at least 3.5 stars; (iii) They are "smart", i.e., with either Wi-Fi or BLE capability which can be used to connect to a smartphone.

Table 1 summarizes the devices, their wireless types, and activities conducted in our experiments. Note that devices with "Wi-Fi or Bluetooth" typically use Bluetooth to communicate with the phone and only use Wi-Fi for initial set up. The only exception is Withings Sleep Analyzer, described as "Wi-Fi & Bluetooth", which uses Wi-Fi for communication and Bluetooth for setup.

3.2. Data Collection

Active Man-in-the-Middle (MITM) attack is an active cybersecurity attack where a third party end can eavesdrop and stalk the connection [4]. GATTacker [13] implemented the BLE MITM attack and was first proposed by Slawomir Jasek in 2016. Specifically, GATTacker creates two fake BLE entities to impersonate and connects with their real counterparts. While the real central and peripheral believe they are talking directly with each other, the fake entities eavesdrop and log the BLE communication data. GATTacker is an open-source project that is implemented in JaveScript and requires noble [17] and bleno [16] packages. We adapted GATTacker [13] to our study by using two Raspberry Pi - each for one fake entity. We captured BLE traffic directly on the Raspberry Pis.

Device	Wireless Type	Health Functionality
Apple Watch	WiFi or Bluetooth	Breath Blood O2 ECG Cycle
Fitbit Inspire	WiFi or Bluetooth	Running Relax Swimming Synchronization
Mi Band	WiFi or Bluetooth	Running Walking
Withings BPM	WiFi or Bluetooth	Blood Pressure
Withings Sleep Analyzer	WiFi & Bluetooth	Sleep Quality
Withings Body Scale	WiFi or Bluetooth	Body Weight
iHealth Blood Pressure	Bluetooth	Blood Pressure
SonoHealth EKG Monitor	Bluetooth	EKG
HealthTree Blood Oxygen Monitor	Bluetooth	Blood Oxygen
Govee Thermometer Hygrometer	Bluetooth	Temperature
Pip Stress Level Monitor	Bluetooth	Stress Level
Fitdays Weight Scale	Bluetooth	Body Weight
FitIndex Body Fat Scale	Bluetooth	Body Weight

Table 1: Table of devices, its wireless type, and activities.

4. BLE Active Attack

4.1. Attack Overview

In this section, we explore active BLE attacks that can (i) perform Denial-of-Service attack to prevent the device from connecting to the real smartphone, and/or (ii) extract sensitive health data from the encrypted communication packets sent by the device. Note that the active attacker does not compromise any devices or pairing keys, which is the same as in the passive attacks. Instead, the active attacker poses itself as the Man-in-the-Middle (MITM) and creates separate connections with the device and the phone, respectively. Consequently, both the device and the phone mistakenly believe that they are talking to each other directly. The MITM attacker can then observe all data sent by the device, and even inject or alter the data before forwarding them to the phone.

MITM Tool. We use the GATTacker [13] tool to implement the MITM attacks on BLE connections between the device and the phone. We create fake middle entities that sit in between victim devices. The fake entities pretend to be the real "central device" (e.g., phone) and communicate

Figure 1: BLE Man-in-the-Middle (MITM) Attack

with other victim devices ("peripheral" devices, which are restricted to one BLE connection at a time). The attack flow is shown in Figure 1.

In our attack experiments, two Raspberry Pis are used to resemble the fake entities. The fake central and peripheral devices connect with the real counterparts, pass information, log, and steal data from the conversation. We use the Oximeter as an example to outline the detailed steps as follows:

- The fake central device (X) advertises its presence, collects the advertising packets and service information from the Oximeter (B).
- The fake central (X) connects with the fake peripheral (Y) through HTTP Proxy and sends the cloned information to Y.
- The fake peripheral (Y) connects to the real central (A) using the cloned information and tricks A to believe Y as the real Oximeter.
- The fake central (X) connects to the Oximeter and tricks the device to believe it is the real phone.
- As A and B believe they connect to each other, they will communicate and transmit data. Their exchanged data is then captured by X and Y.

Implementation. We implemented GATTacker on a Raspberry Pi with two Bluetooth CSR 8510-based USB dongles. BLENO [16] and NOBLE [17] with Node 8.11.1 were used to configure the Pi, and the two dongles were used as fake central and fake peripheral. Fake central and fake peripheral communicated with each other using WebSocket. Hook functions were used for data interception and manipulation.

We perform the follow attacks using our testbed:

- **Denial of Service**: a BLE peripheral device only connects with one central device — once the connection is established, the peripheral device stops advertising itself. Thus, another central device is not able to discover it. In the MITM attack, once the fake central device connects with the peripheral device, other central devices cannot connect. This will cause DoS attack, which affects the availability of the peripheral device.
- **Data Interception**: because the fake devices establish connections directly with the device and the phone, they can observe data content inside the encrypted packets.

Device	DoS	Data Interception
Apple Watch		
Fitbit Inspire		
Mi Band		
Withings BPM		
Withings Sleep Analyzer		
Withings Body Scale		
iHealth Blood Pressure Monitor	△	●
SonoHealth EKG Monitor	△	●
HealthTree Blood Oxygen Monitor	△	●
Govee Thermometer Hygrometer	△	
The Pip Stress Level Monitor	△	
Fitdays Weight scale		
FitIndex Body Fat Scale		

Table 2: Device vunlerability to MITM attacks.

4.2. Evaluation

We performed the MITM attacks against all wearable devices in our dataset. Table 2 shows the vulnerability of devices to DoS and data interception attacks.

DoS Attack. Due to the power conservation characteristic of BLE connection, peripheral devices typically are only able to connect with one central device. A MITM attack is able to conduct a DoS attack thanks to its ability to generate a fake central device. The generated fake central device broadcasts its information with higher frequency, attempting to connect with the peripheral device. Once connected, the peripheral cannot connect and communicate with the real central device, leading to a DoS attack.

Data Interception. MITM attacks may uncover the encrypted data sent by devices in the BLE communication packets. The packets have the following structure: {**Timestamp — Type — Service UUID — Characteristic UUID — Hex data**}. We successfully parse the Hex data format for three devices, but fail to do so for the other two. Figure 2 illustrates an example of the captured data from the Oximeter at the MITM attacker. The first two bits of data in red are the blood oxygen saturation levels (SpO2) in Hex. The following two bits in red are the pulse rate (PR) in Hex. The Oxygen reading in Figure 2 are SP02: 99, PR: 75; SP02: 99, PR: 74; SP02: 99, PR: 73. The attacker can then simply convert Hex values to Decimal values in human-readable formats. This example demonstrates that the active attacker can directly see the sensitive health measurement data from the user. Such attacks do not require any prior knowledge or training, and can be performed using simple tools as we showed in our experiments.

Data Manipulation. After intercepting the traffic, depending on the encryption of the communication, the malicious entity can alter the original data and relay. This may cause serious problems, such as sending the wrong measurement to the user. We plan to explore such attack in greater details in future work.

Takeaway. Active MITM attacks can intercept the BLE connection between the device and the phone, enabling the adversaries to launch DoS attacks or extract encrypted user health data which cannot be observed by a passive attacker.

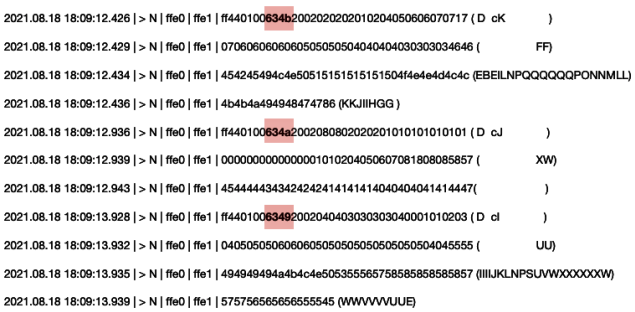

Figure 2: Data Eavesdropped of Oximeter

5. BLE Security Discussion

We discuss existing BLE security mechanisms that can prevent major attacks if correctly implemented on the devices.

5.1. Security Pairing

There are two pairing processes designed for BLE devices: (i) *Legacy Pairing* mode for devices operating on BLE versions 4.0 and 4.1, and (ii) *Secure Connection* mode for devices operating on BLE version 4.2 and beyond. In *Legacy Pairing* mode, the devices exchange a Temporary Key (TK) and employ it to generate a Short Term Key (STK) responsible for encrypting the connection. While in *Secure Connection* mode, the devices use a single Long Term Key (LTK) to encrypt the communication by utilizing Elliptic Curve Diffie-Hellman (ECDH) for key exchange and authentication. Notably, there are four major pairing strategies as we discussed below:

- *Just Works* does not require any user input to authenticate the devices. IoT devices without any I/O capabilities, such as headphones, sensors, light bulbs, use this method for pairing. In *Legacy Paring*, the TK is hard coded to 0, while in *Secure Connection*, a random seed is employed during the key generation process.
- *Passkey Entry* requires that both connecting devices should have input and display output capabilities. In both modes, a randomly generated numeric passkey will appear on one device (i.e., phone), and the user needs to input the pin on the other device to pass the authentication.
- *Out-of-Band* requires both connecting devices to support a different wireless communication method, like NFC, to exchange secret data, such as the TK in *Legacy Pairing* and the public key in *Secure Connection* mode.
- *Numeric Comparison* is exclusively accessible in BLE *Secure Connection* mode, and requires user input for authentication. Following the key exchange process, both devices generate and display a six-digit number, and the user must select the matching number on the device to confirm the connection.

Just Works is the most commonly adopted pairing mode in the IoT environment due to the absence of input/output capabilities in many IoT devices, such as light bulbs, smart locks, and heart rate monitors [29]. Nonetheless, Just Works remains susceptible to various security threats. In *Legacy Pairing* mode, the presence of a hard-coded Temporary Key renders it remarkably easy for attackers to launch brute-force attacks on the Short Term Key (STK), facilitating unauthorized decryption of communication. Just Works remains vulnerable to MITM attacks even with *Secure Connection* as it lacks mechanism to verify the authenticity of the connection. Passkey Entry also remains vulnerable to active MITM attacks, particularly when the attacker can sniff all the pairing packets to brute force the temporary key. Out-of-Band (OOB) pairing offers protection against passive eavesdropping and MITM attacks when the key exchange OOB channel is secure. However, such pairing mechanism requires the IoT devices to support a different wireless communication method that may not be applicable for many IoT devices. Numeric Comparison pairing necessitates manual confirmation value checks on both connecting devices, providing defense against MITM attacks. Nonetheless, this method requires that both IoT devices possess input/output capabilities.

5.2. MAC address randomization

This is another security mechanism to prevent using MAC Address as an identifier for passive sniffing. Bluetooth MAC address (BD_ADDR) is a 48-bit identifier uniquely for Bluetooth devices. Bluetooth MAC address is a public address that is globally fixed and must be registered with the IEEE. Public address cannot be renewed, which leaves the possibility of device tracking. To enhance communication privacy, BLE supports other 3 types of random addresses besides the public address:

- Random Static device address. This address is a popular alternative to Public Address since there are no fees in using it. It is programmed into the device, which either cannot be changed or can be renewed at bootup. It cannot be renewed during runtime.
- Random Resolvable device address. The following two types of addresses are used for hiding the identity and preventing tracking. Resolvable Random Private address can be resolved by a trusted (bonded) device using a shared key, the Identity Resolving Key (IRK). It preserves privacy while allowing trusted parties to identify the BLE devices.
- Random Non-resolvable device address. A randomly generated address that can be renewed at any time and is not resolvable by any devices.

Despite the existing security mechanisms, BLE still suffers from security and privacy issues because many IoT devices fail to implement these mechanisms properly.

The Bluetooth Core Specification [12] recommends that the Random Non-resolvable and Random Resolvable addresses shall be generated frequently to preserve privacy. The recommended renewal time is 15 minutes. However, Celosia *et al.* [23] identified that a significant number of

BLE devices failed to follow this specification. In their experiments, 6% of Random Resolvable and 4% of Random Non-resolvable addresses have a lifetime larger than 15 minutes and up to 69 days. Furthermore, the authors showed that the advertising packets contain unique identifiers which do not reset after address randomization. Examples of the identifiers are device names, service UUID, etc. These identical identifiers can be used for tracking and eavesdropping.

Our attack experiments also verify that at least five wearable devices in our dataset are vulnerable to active MITM attacks due to insecure pairing mechanisms. Hence, we urge that IoT vendors, especially for healthcare devices which involve sensitive health information, to follow BLE best practices and adopt security mechanisms to prevent both passive and active attacks.

6. Conclusion

In this work, we perform a preliminary study on the vulnerability of Bluetooth connections from top wearable health devices. We demonstrated an active adversary that can eavesdrop and decode the actual measurement data. We discussed the defenses of using existing BLE security mechanisms. Our work highlights the security privacy risk of BLE communication and emphasizes the importance of adopting better BLE privacy countermeasures.

References

[1] Bluetooth. https://www.bluetooth.com/.

[2] Bluetooth Low Energy (BLE). https://www.bluetooth.com/learn-about-bluetooth/tech-overview/.

[3] Long Range Bluetooth Audio Transmitter. https://www.amazon.com/Miccus-Bluetooth-Transmitter-Receiver-Headphones/dp/B075J4RKGH.

[4] Man in the Middle (MitM) Attack. https://en.wikipedia.org/wiki/Man-in-the-middle_attack.

[5] Bluetooth Low Energy for Bluetooth in Smart Wearable Technology. https://developex.com/blog/bluetooth-low-energy-for-bluetooth-in-smart-wearable-technology/, 2017.

[6] Carriers selling your location to bounty hunters: it was worse than we thought. https://www.theverge.com/2019/2/6/18214667/att-t-mobile-sprint-location-tracking-data-bounty-hunters, 2020.

[7] Data Onboarding 101—Match Rates. https://liveramp.com/our-platform/data-onboarding/, 2020.

[8] Google can still use Bluetooth to track your Android phone when Bluetooth is turned off. https://qz.com/1169760/phone-data/, 2020.

[9] TfL introduces wifi tracking to improve ads. https://www.thedrum.com/news/2019/05/22/tfl-introduces-wifi-tracking-improve-ads, 2020.

[10] The rise of employee health tracking. https://www.bbc.com/worklife/article/20201110-the-rise-of-employee-health-tracking, 2020.

[11] Wearable technology in health care: Getting better all the time. https://www2.deloitte.com/xe/en/insights/industry/technology/technology-media-and-telecom-predictions/2022/wearable-technology-healthcare.html, 2021.

[12] Bluetooth Core Specification. https://www.bluetooth.com/specifications/bluetooth-core-specification/, 2023.

[13] GATTacker. https://github.com/securing/gattacker, 2023.

[14] How To Increase Bluetooth Range. https://thegadgetbuyer.com/how-to-increase-bluetooth-range/, 2023.

[15] Amazon.com. https://www.amazon.com/, 2024.

[16] bleno, A Node.js module for implementing BLE (Bluetooth Low Energy) peripherals. https://github.com/noble/bleno, 2024.

[17] noble, A Node.js BLE (Bluetooth Low Energy) central module. https://github.com/noble/noble, 2024.

[18] Abbas Acar, Hossein Fereidooni, Tigist Abera, Amit Kumar Sikder, Markus Miettinen, Hidayet Aksu, Mauro Conti, Ahmad-Reza Sadeghi, and Selcuk Uluagac. Peek-a-boo: I see your smart home activities, even encrypted! In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2020.

[19] Marwan Ali Albahar, Keijo Haataja, and Pekka Toivanen. Bluetooth mitm vulnerabilities: a literature review, novel attack scenarios, novel countermeasures, and lessons learned. *International Journal on Information Technologies & Security*, 8(4):25–49, 2016.

[20] Noah Apthorpe, Danny Yuxing Huang, Dillon Reisman, Arvind Narayanan, and Nick Feamster. Keeping the Smart Home Private with Smart (er) IoT Traffic Shaping. In *Privacy Enhancing Technologies Symposium (PETS)*, 2019.

[21] Noah Apthorpe, Dillon Reisman, Srikanth Sundaresan, Arvind Narayanan, and Nick Feamster. Spying on the smart home: Privacy attacks and defenses on encrypted IoT traffic. *arXiv preprint arXiv:1708.05044*, 2017.

[22] Ludovic Barman, Alexandre Dumur, Apostolos Pyrgelis, and Jean-Pierre Hubaux. Every byte matters: Traffic analysis of bluetooth wearable devices. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2021.

[23] Guillaume Celosia and Mathieu Cunche. Saving private addresses: An analysis of privacy issues in the bluetooth-low-energy advertising mechanism. In *EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2019.

[24] Rajarshi Roy Chowdhury, Sandhya Aneja, Nagender Aneja, and Emeroylariffion Abas. Network traffic analysis based iot device identification. In *Proceedings of the International Conference on Big Data and Internet of Things*, 2020.

[25] Aveek K Das, Parth H Pathak, Chen-Nee Chuah, and Prasant Mohapatra. Uncovering privacy leakage in BLE network traffic of wearable fitness trackers. In *Proceedings of the 17th international workshop on mobile computing systems and applications*, 2016.

[26] Chenxin Duan, Shize Zhang, Jiahai Yang, Zhiliang Wang, Yang Yang, and Jia Li. Pinball: Universal and

robust signature extraction for smart home devices. In *2021 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2021.

[27] Andrew Hintz. Fingerprinting websites using traffic analysis. In *International workshop on privacy enhancing technologies*. Springer, 2002.

[28] PETER KLERKS. The network paradigm applied to criminal organisations: theoretical nitpicking or a relevant doctrine for investigators? *Transnational organised crime*, pages 111–127, 2004.

[29] Karim Lounis and Mohammad Zulkernine. Bluetooth low energy makes "just works" not work. In *IEEE cyber security in networking conference (CSNet)*, 2019.

[30] Liming Lu, Ee-Chien Chang, and Mun Choon Chan. Website fingerprinting and identification using ordered feature sequences. In *European Symposium on Research in Computer Security (ESORICS)*, 2010.

[31] Thomas Martin, Michael Hsiao, Dong Ha, and Jayan Krishnaswami. Denial-of-service attacks on battery-powered mobile computers. In *Second IEEE Annual Conference on Pervasive Computing and Communications, 2004. Proceedings of the*, pages 309–318. IEEE, 2004.

[32] Jonathan Muehlstein, Yehonatan Zion, Maor Bahumi, Itay Kirshenboim, Ran Dubin, Amit Dvir, and Ofir Pele. Analyzing HTTPS encrypted traffic to identify user's operating system, browser and application. In *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, 2017.

[33] Jeffrey Pang, Ben Greenstein, Ramakrishna Gummadi, Srinivasan Seshan, and David Wetherall. 802.11 user fingerprinting. In *Proceedings of the ACM international conference on Mobile computing and networking (MobiCom)*, 2007.

[34] F Stajano and R Andreson. The resurrecting duckling: Security issues for ad-hoc wireless networks, security protocols, 7th. In *International Workshop Proceeding, Lecture Notes in Computer Science*, 1999.

[35] Da-Zhi Sun, Yi Mu, and Willy Susilo. Man-in-the-middle attacks on secure simple pairing in bluetooth standard v5. 0 and its countermeasure. *Personal and Ubiquitous Computing*, 22:55–67, 2018.

[36] Rahmadi Trimananda, Janus Varmarken, Athina Markopoulou, and Brian Demsky. Packet-level signatures for smart home devices. In *Network and Distributed Systems Security (NDSS) Symposium*, 2020.

[37] Jason Uher, Ryan G Mennecke, and Bassam S Farroha. Denial of sleep attacks in bluetooth low energy wireless sensor networks. In *MILCOM 2016-2016 IEEE Military Communications Conference*, pages 1231–1236. IEEE, 2016.