future internet

MDPI

Article

25 Years of Bluetooth Technology

Sherali Zeadally [1,*], Farhan Siddiqui [2] and Zubair Baig [3]

[1] College of Communication and Information, University of Kentucky, Lexington, KY, 40506, USA
[2] Department of Mathematics and Computer Science, Dickinson College, Carlisle, PA 17013, USA
[3] School of Information Technology, Deakin University, Geelong 3216, Victoria, Australia
* Correspondence: szeadally@uky.edu

Received: 12 August 2019; Accepted: 2 September 2019; Published: 9 September 2019

Abstract: Bluetooth technology started off as a wireless, short-range cable replacement technology but it has undergone significant developments over the last two decades. Bluetooth radios are currently embedded in almost all computing devices including personal computers, smart phones, smart watches, and even micro-controllers. For many of us, Bluetooth is an essential technology that we use every day. We provide an insight into the history of Bluetooth and its significant design developments over the last 25 years. We also discuss related issues (including security) and Bluetooth as a driving technology for the Internet of Things (IoT). Finally, we also present recent research results obtained with Bluetooth technology in various application areas.

Keywords: bluetooth; internet of things; low-energy; mesh; networking; protocol; security

1. Introduction

The Bluetooth radio technology was developed by L. M. Ericsson in 1994. The standard is named after the King of Denmark, Harald Blaatand ("Bluetooth"). Major mobile phone manufacturers and technology providers comprising IBM, Nokia, Intel, Ericsson, and Toshiba created the Bluetooth Special Interest Group (SIG). The aim of the group was to invent an open specification for wireless technologies of short range. Bluetooth SIG continues to oversee the Bluetooth technology today. 3COM, Microsoft, Lucent, and Motorola also promote the SIG group currently. The SIG [1] now has more than 1900 companies since it was created. Classic Bluetooth (which was the original specification of Bluetooth) was intended to transmit data wirelessly between computer devices. Over the last twenty years of SIG existence, Bluetooth technology has constantly been able to meet the growing demands for wireless innovation. Since its introduction in 1998, Bluetooth device shipments continue to increase without the slightest expectation of a decrease in demand. Figure 1 shows the previous and predicted annual Bluetooth device shipments [2].

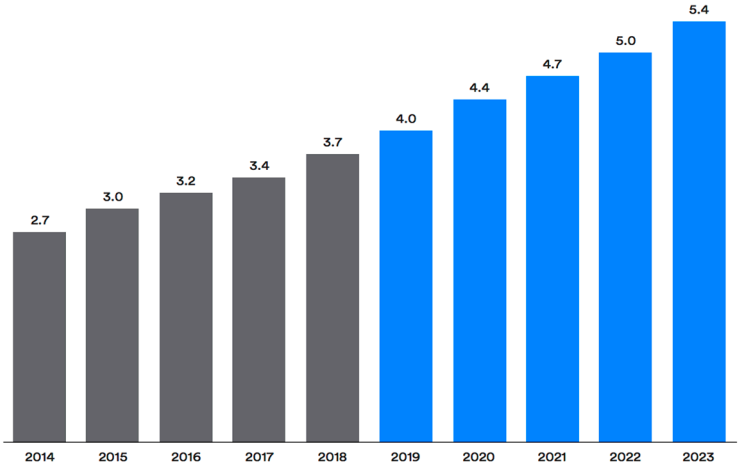

Figure 1. Annual Bluetooth device shipments (in billions) [2].

Main contributions of this work

We summarize the main contributions of this work as follows:

- We present the evolution of Bluetooth technology over the last two decades as well as current and future trends of Bluetooth.
- We describe internetworking solutions between Bluetooth and IP-based networks. We also present the communication protocols used between Bluetooth and Zigbee.
- We discuss recent attacks on Bluetooth and solutions to mitigate them.
- Finally, we highlight state-of-the-art research efforts that have been leveraging Bluetooth technology to provide innovative solutions in areas such as health, location tracking, and smart homes.

Bluetooth Low Energy, generally referred to as "Bluetooth Smart" was specifically created for scenarios, which have a low duty cycle and was marketed to the public in October 2010. For instance, a belt that records the heart rate can be hooked on for many hours, say during a workout session, but will only need to send limited number of bytes of data per second. Therefore, in an enhanced and optimized protocol, Bluetooth radios will be kept "on" for just a few milliseconds. In comparison, a headset or a wireless speaker would transmit much more data (typically many kilobytes in just one second and the radio is turned on for a reasonably longer time.

Bluetooth Low Energy (BLE) was devised specifically for use cases that Bluetooth classic was not suited for. Currently, Bluetooth Smart has been broadly adopted in many types of health and fitness applications including some sports-related use cases. The BLE technology also has potential for use in medical devices and for newer applications such as beacons and proximity tags. Recently there is also an increase in many types of "connected" devices for home networking and smart grids, as well as an array of smart watches, motion monitoring devices and other smart tools and gadgets.

Bluetooth Smart serves as a driver of the Internet of Things (IoT). Instead of acquiring a direct Internet connection, gadgets that are Bluetooth Smart may acquire Internet connectivity via other devices that are Bluetooth Smart Ready. This approach provides a cheaper and lower power consuming solution [3]. It is predicted that by 2023, more than 1.6 billion Bluetooth Low Energy devices will be shipped each year, and 90% of all Bluetooth devices will include Bluetooth Low Energy technology [2]. Figure 2 shows the Bluetooth device shipment numbers by the radio version.

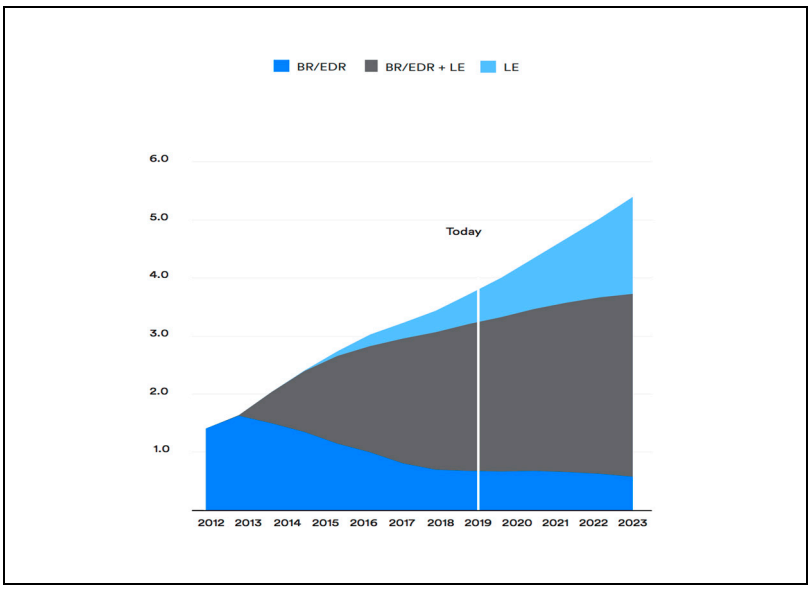

Figure 2. Bluetooth device shipments by the radio version (in billions) [2].

Today, we have three main versions of Bluetooth technology, namely: a) Bluetooth Basic Rate (BR)/Enhanced Data Rate (EDR), b) Bluetooth Low Energy (BLE), and c) Bluetooth Mesh. Figure 3 shows the timeline for Bluetooth evolution. The characteristics relevant to each Bluetooth version are shown in Table 1 below. The rest of this paper is organized as follows. Section 2 discusses the three Bluetooth technologies that exist today. Section 3 describes how Bluetooth devices can be connected to IP networks. Section 4 describes the communication of Bluetooth radios with Zigbee. Section 5 highlights some of the most common security threats associated with the use of Bluetooth technology while Section 6 highlights the latest research efforts on Bluetooth technology. Finally, we make some concluding remarks in Section 7.

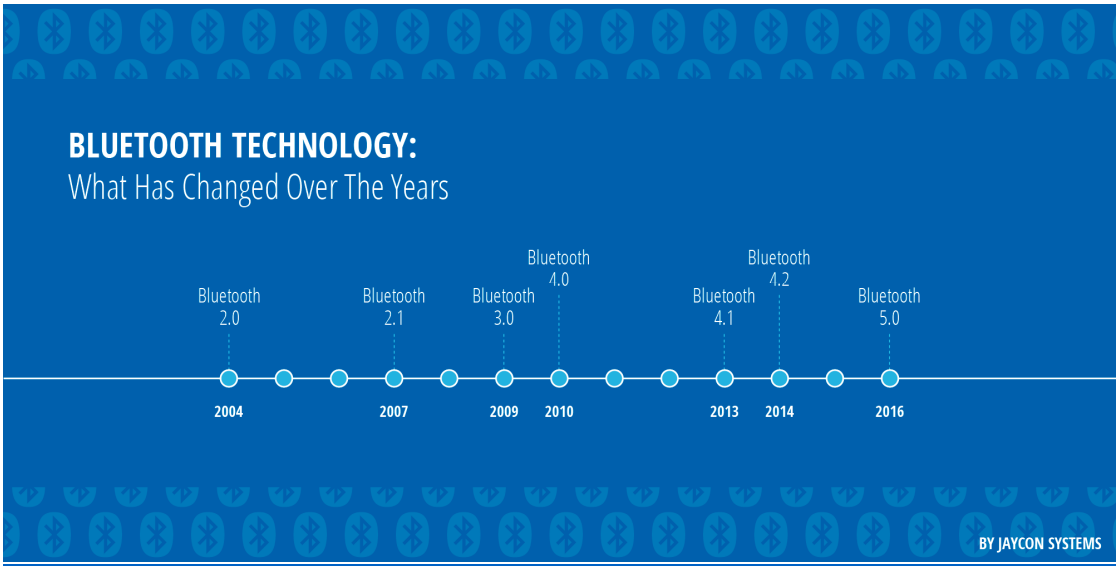

Figure 3. Evolution of Bluetooth technology (2004–16) [4].

Table 1. Bluetooth characteristics [5].

Bluetooth Specification	v1.1	v2.0 + EDR	v2.1 + EDR	v3.0 + HS	v4.0 + LE	v4.1	v4.2	v5.0
Year	2002	2004	2007	2009	2010	2013	2014	2016
Basic Rate	YES	YES	YES	YES	YES	YES	YES	YES
Enhanced Data Rate (EDR)	NO	YES	YES	YES	YES	YES	YES	YES
High Speed (HS)	NO	NO	NO	YES	YES	YES	YES	YES
Low Power (LE)	NO	NO	NO	NO	YES	YES	YES	YES

2. Bluetooth Technologies

This section describes the operation of different versions of Bluetooth technologies and their respective applications.

2.1. Bluetooth BR/EDR

The Bluetooth system operates in the 2.4 GHz Industrial, Science and Medical (ISM) band. The frequency band is in the 2400.0–2483.5 MHz range, with Radio Frequency (RF) channels being spaced at 1 MHz. Two data transmission modes are defined: Basic Rate (BR) that uses a shaped, binary Frequency Modulation (FM) to minimize transceiver complexity, and Enhanced Data Rate (EDR) that uses Phase Shift Keying (PSK) modulation and has two further variants: $\pi/4$-Differential Quadrature Phase Shift Keying (DQPSK) and Differential Phase Shift Keying (DPSK). Basic Rate (BR) mode is a mandatory part of the Bluetooth specification. The modulation that BR uses is Gaussian Frequency Shift Keying (GFSK). In Enhanced Data Rate (EDR) mode, the scheme used for modulation is altered inside the packet. The packet header and the access code are sent using the Basic Rate of 1 Mbit/s GFSK modulation scheme, while subsequent synchronization sequence, payload, and trailer sequence are transmitted with the EDR PSK modulation scheme [6].

A combination of devices connected via Bluetooth in an ad-hoc mode is called a piconet. In a piconet, one Bluetooth device acts as a master and the other devices act as slaves. The device that starts communication is the master node and the other devices are slaves. A piconet consists of a minimum of two connected devices, and can extend up to eight connected devices (one master and seven slaves). Every Bluetooth device can act as a master or a slave device. When there is a single slave device, a piconet is a simple point-to-point link. A point-to-multipoint arrangement can have a maximum of seven active slaves under the control of a single master. Figure 4 shows the two different configurations of a Bluetooth piconet. Slaves always exchange data through the master node. Communication across piconets creates a scatternet. Scatternets exist when a Bluetooth device is a slave in one piconet and simultaneously a master or a slave in another piconet.

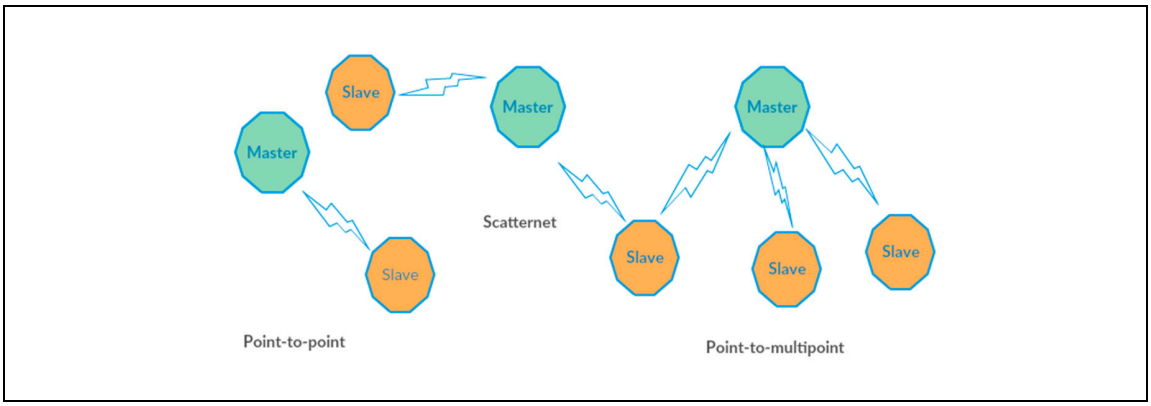

Figure 4. Bluetooth piconet configurations [1,7].

All devices share the Bluetooth master's clock. The basic clock has a clock cycle time of 312.5 microseconds (µs). The time base of packet exchange is the basic clock. A slot of 625 microseconds consists of two clock cycles; a slot pair of 1250 µs consists of two slots. The master sends in even slots and receives in odd slots; for the case of slaves, transmission occurs in odd slots and reception occurs in even slots. Upon powering up, a Bluetooth device can operate in slave mode when the master device is already operating. The slave waits for inquiries from the masters and provides replies. The master and slave devices can change roles, which would be necessary when a Bluetooth device participates in more than one piconet.

2.1.1. Bluetooth Protocols

Bluetooth operation utilizes seven different protocols: radio protocol, baseband protocol, Radio Frequency Communication (RFCOMM), the Service Discovery Protocol (SDP), Link Management Protocol (LMP), the Logical Link Control and Adaptation Layer Protocol (L2CAP), and the Host Controller Interface (HCI) protocol. Figure 5 illustrates the Bluetooth protocol stack.

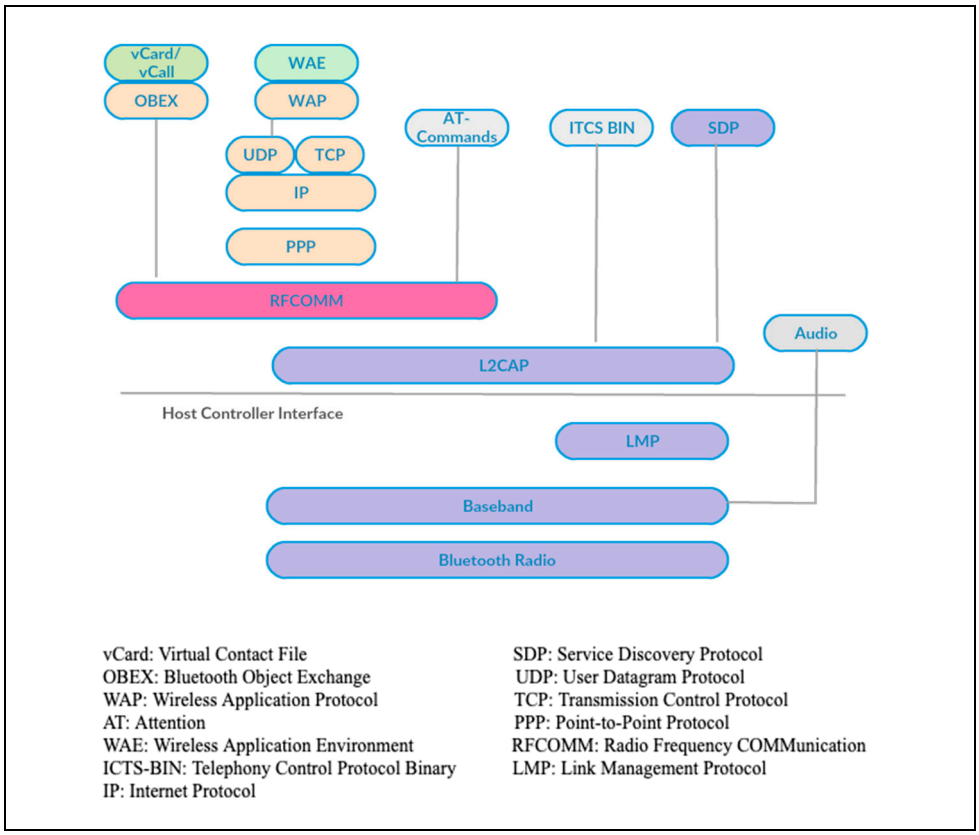

Figure 5. Bluetooth protocol stack [1,7].

The radio protocol runs in the digital radio-processing component of the Bluetooth radio. Bluetooth utilizes the GFSK modulation. The Bluetooth ISM band is divided into 79 channels of 1 MHz each. Each channel is further divided into 625 microsecond timeslots which results in 1600 slots per second. Data is transmitted through these slots and channels. The baseband protocol processes the sent and received signals. The handling of error correction, links, packets, and flow control is implemented by the baseband protocol.

Bluetooth enabled devices can establish Synchronous Connection-Oriented (SCO) and Asynchronous Connectionless links (ACL). Data packets use ACL primarily while voice packets use SCO. ACL links are primarily defined for data transmission. They support asymmetrical, symmetrical and packet-switched connections. Multi-slot packets use the ACL link type. With ACL,

the maximum achievable data rate is 723 kbps in one direction and 57.6 kbps in the other direction. The master device controls ACL link to allocate bandwidth for a piconet slave. The ACL link supports broadcast messaging from the master node to all slaves in the piconet.

SCO links support symmetrical, point-to-point connections and are mainly used for voice transmissions. For a SCO link, every sixth slot is reserved for a transmitting channel and reservation of a subsequent slot is done for the receiving channel. After the establishment of the connection, both master and slave units send SCO packets when they need to do so. Data and voice transmissions are allowed in one SCO packet type with retransmission of the data portion carried out when the packet gets corrupted.

The Host Controller Interface (HCI) provides a command interface to the baseband controller and link manager. It also provides access to control registers and hardware status. HCI is accessible at the host, host controller, and the transport layer. L2CAP is structured over the baseband protocol which is present in the data link layer. L2CAP segments packets for Bluetooth transmission and when the packet passes through the L2CAP of another Bluetooth device, its original form is restored through reassembly. L2CAP supports only ACL and not the SCO links.

RFCOMM is a transport protocol which allows RS232 serial port emulation over the L2CAP protocol. The Service Discovery Protocol (SDP) uses a request-response message style such that each transaction has a response Protocol Data Unit (PDU) and one request PDU [1,7].

2.1.2. Applications of Bluetooth BR/EDR (Classic Bluetooth)

Classic Bluetooth devices are those that maintain a high-throughput connection. In the future, many computing machines are likely to be BLE enabled. However, specific device categories requiring high data throughput will continue to exist. One example of one such device is the Bluetooth headset where Bluetooth Low Energy does not provide much benefit. Such devices will probably remain classic Bluetooth devices in the foreseeable future [3]. Other examples of classic Bluetooth devices include Bluetooth-enabled car stereos that allow wireless transmission of audio and therefore enables hands-free communication, and wireless keyboards that can be paired up with a smart phone, or a laptop. Bluetooth is also often used for implementing file transfers between any two Bluetooth enabled devices (such as a PC and an iPhone). Furthermore, tethering via Bluetooth allows any 3G/4G enabled device to act as a hotspot and provide Internet access to nearby Bluetooth enabled devices.

2.2. Bluetooth Low Energy

Bluetooth Low Energy (BLE) PHYsical layer (PHY) is a reduced and optimized version of Bluetooth BR PHY. While the BR PHY hops over 79 channels (with the possibility of reducing to 20 channels) and implements discovery over 32 channels, the BLE PHY takes 37 channels, while it performs device discovery on three channels. Since BLE goes via fewer channels while performing discovery, efficiency is achieved, resulting in quick connection establishment (compared to Bluetooth BR/EDR). BLE spacing of channels is 2 MHz in contrast to BR's 1 MHz, and this reduces demands on RF filtering.

BLE connections are identical to sniff sub-rating mode in BR. Therefore, BLE has an energy-efficient method of keeping connectivity while minimizing active radio usage. BLE allows Integrated Circuit (IC) vendors to perform optimizations that are otherwise challenging with BR/EDR Bluetooth. These optimizations allow single-mode chips to be more energy efficient than dual-mode or classic chips. Comparatively, BLE profiles are layered over Generic Attribute Profile (GATT), using the GATT/ATT protocol. In contrast, Bluetooth BR/EDR profiles generally state their protocols, which provides better adaptability, however, makes the implementation more complex [3].

There are three layers in the BLE stack: Controller, Host, and Application. The Physical and the Link layer comprise the Controller. The upper layer functionalities, the Logical Link Control and Adaptation Protocol (L2CAP), the Attribute Protocol (ATT), the Generic Attribute Profile (GATT), the Security Manager (SM) and the Generic Access Profile (GAP) are located in the Host. Host Controller Interface (HCI) enables communication between the Host and the Controller. Applications

reside above the Host layer. Figure 6 illustrates the BLE protocol stack. The L2CAP layer performs segmentation; reassembly of packets. Device communication is defined by the ATT and the SM implements authentication, pairing and key distribution.

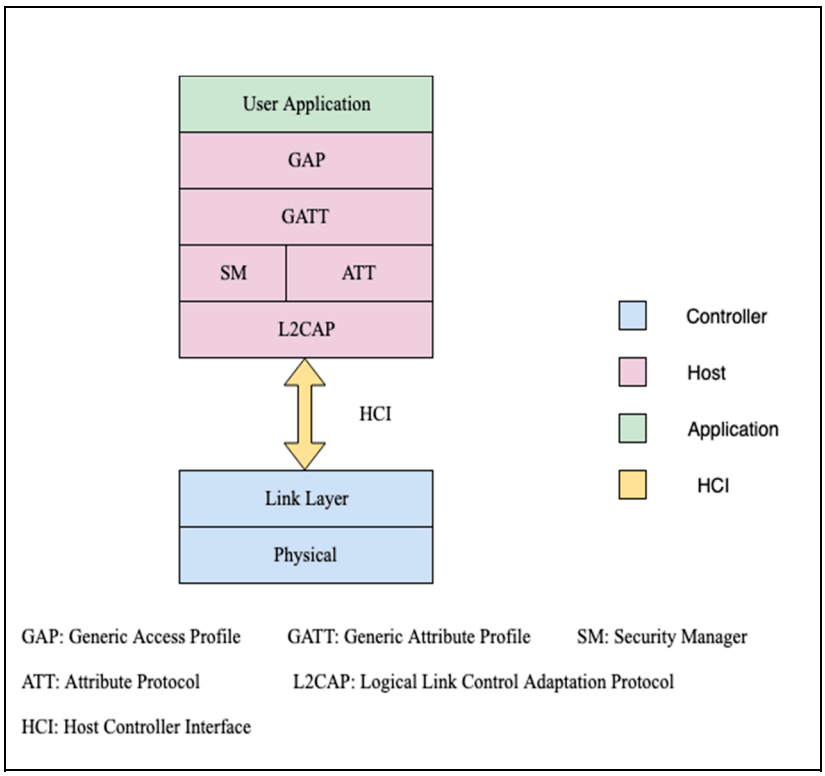

Figure 6. Bluetooth Low Energy protocol stack [8]. Reproduced with permission from Dian et al., IEEE 2018.

Data to be transmitted is kept in a GATT server database. This data can be relocated to the GATT client. The application dictates whether the GATT server/client is kept on the master or the client node. GATT specifies a framework for communication between GATT server and client databases.

Based on the hierarchy ({*profile-service, characteristics-attributes*}), the information is organized on the GATT server. Attributes may be defined as small, addressable GATT data entities.

A service includes features and is characterized via specific characteristics. Service groups are described using a *profile*. GAP identifies ways of discovering other BLE devices and describes four functions: advertiser, scanner or initiator, master and slave. Advertising, scanning, and initiating modes are defined in the discovery process to search BLE devices. The scanner searches only for the advertiser while the initiator requests a connection with the advertiser [8].

2.2.1. BLE Data Exchange Methods

There exists, a few methods by which a GATT server is accessed by a client: read, write, notify and indicate. Using read and write, the GATT client can read and write the characteristic values from the GATT server. Notify is used to notify the GATT client when a characteristic value has been updated by the GATT server. Indicate is similar to notify with notification acknowledgement to GATT server. Notifications or indications or explicit read requests can be used to move data from GATT server to client [8].

2.2.2. Applications of Bluetooth Low Energy (Bluetooth Smart)

BLE has numerous applications, some of which are discussed below:

In GATT-based devices and services, devices are connected to a Bluetooth Smart Ready device called a gateway. It is worthwhile noting that a Bluetooth Smart Ready device is one that supports both classic Bluetooth as well as Bluetooth Smart radios. The devices implement one or more GATT-based services, offered to the GATT-based devices. The gateway interprets the service / profile and then exposes an "xyz" Internet Application Programming Interface (API) (e.g., a RESTful API) or contains an "xyz" application that sends and receives data from the Internet to the GATT-based BLE device. In this use case, the gateway can be a dedicated fixed device or a portable device. The use case can be applied in, for instance, connected sensors in a building (home automation or home care), body-worn health or fitness sensors and various metrology or industrial devices, where the centralized gateway would facilitate communication.

BLE Beacons

In 2013, Apple introduced a new technology, called iBeacon. iBeacon facilitates internal positioning of devices. iBeacons can be used by Android systems. A device can get push notification from nearby iBeacon devices. iBeacon works with Bluetooth Low Energy (BLE), also known as Bluetooth 4.0 or intelligent Bluetooth, and can enable us to determine our position in a store. The iBeacon system is capable of transmitting announcements regarding items that may be useful to us in the store.

The Internet of Things (IoT) refers to uniquely identifiable objects and their virtual representations in an Internet-like structure [9]. iBeacon uses the low energy Bluetooth technology that can register the nearness of an object and send an unambiguous universal identifier token from a compatible application to transform it into a physical location. The beacons come in different formats, including small coin cell powered devices, flash drives, and software in embedded devices such smart-bulbs, and so on.

BLE beacon uses can be classified into one of the following categories [10]:

- Static Point of Interest (POI): Static POI refers to initiating an action when a user enters a specific physical area. For example, the location of the POI could map to a front reception desk or an elevator, etc. Once an individual is at or near a specific location, determining the position or direction of the user allows enhancing the user experience. For instance, a hotel could deploy BLE beacons and build applications for customers that offer the ability to unlock room doors by simply being in close proximity to the door.
- Indoor mapping: Many companies attempted to resolve the indoor geo-location issue and BLE beacons could be a potential solution. By creating applications and installing BLE beacons in big shopping centers, customers can steer through retail shops easily. This added value to customers will encourage them to keep the application open and the Bluetooth radio turned on.
- Two-way proximity: This allows the user device to act as the beacon. An action or event can be triggered on one device that sends the user's ID and micro-location to a peer or administrator application, enabling the capability of locating other people such as friends, etc.
- Analytics: There are many prospects for marketers to collect valuable insights from BLE beacons: getting information such as who entered a specific region, at what time and for how long. Getting this information can provide insight about a specific location. Marketers can use this data to constantly customize offerings in order to instill confidence in their customers [10].

2.3. Bluetooth Mesh

The Bluetooth mesh Standard is a publish/subscribe model in which publishers and subscribers can subscribe to any topic. For instance, electric switches can publish and electric lamps can subscribe to a topic. A mesh node can also subscribe to several addresses but publish to one address. Addresses are generally stored in a subscriber list.

The mesh standard outlines different addresses: unicast and group address. Every node acquires a unique unicast address when it joins the mesh network. A group of nodes is represented by a group

address. A new group address needs to be added to the subscriber list before a node can join that group. Once a node acquires a group address, any other node can send a message to it via its unicast address or its group address. A mesh topology is utilized to connect publishers to subscribers.

The Bluetooth mesh standard enables communication between nodes via scanning and advertising, and flooding mechanisms. Flooding allows nodes in the network to repeatedly relay messages to other nodes until they reach the correct destination. Bluetooth Mesh nodes transmit packets using a random time interval and not a fized interval as in the case of BLE. Advertisement channels are scanned using a 100% duty cycle for receiving incoming packets. Therefore, mesh nodes continuously scan for incoming packets except in time intervals when transmission is occurring. It is worthwhile noting that the BLE advertisement packet used by mesh standard [11] is supported by both, BLE and Bluetooth mesh devices.

The mesh standard [11,12] allows backward compatibility feature to allow BLE devices to join the mesh network. Therefore, native BLE devices can also join a Bluetooth Mesh network. This is accomplished via the proxy feature implemented in a "proxy node". The "proxy node" can communicate either by using the default BLE advertising capability or by utilizing backward compatibility that allows BLE support. In order to prevent the flooding mechanism to cause any scalability issues, Bluetooth mesh utilizes a "relay" feature. Only nodes with the "relay feature" enabled can forward packets in to the mesh network. A Time-to-Live (TTL) field is associated with each message. A message is only relayed if its TTL field is greater than 1.

The mesh standard also defines a Friendship feature which helps implement power conservation. The feature allows devices connected to the main power supply to assist low-power nodes to scan and join the mesh network. The low-power device is the low-power node and the assisting device is the friend node. A friend node stores messages that arrive for its low-power node and relays messages that it gets from its low-power node into the mesh network. Using the friendship feature, low-power nodes need not implement a 100% duty cycle thereby, reducing energy consumption.

2.3.1. Bluetooth Mesh Protocol Stack

Nodes in a Bluetooth mesh network implement the Bluetooth mesh stack (Figure 7) as discussed below:

- *BLE Core Specification:* The standard built on top of the BLE specification utilizes advertising and connection oriented features.
- *Bearer layer:* This layer provides an abstraction (called a "bearer") for the underlying BLE specification for the top layers. Bearers could be Advertising (ADV) bearers or Generic Attribute Profile (GATT) bearers. The ADV bearer abstracts BLE advertising and the GATT bearer abstracts BLE connections.
- *Network layer:* This layer allows relaying and security.
- *Transport layer:* Segmenting and reassembly of messages are the primary responsibilities of this layer.
- *Access layer:* This layer is an interface between application focused layers and the layers below. The access layer enables correct exchange of messages between the layers above and below.
- *Foundation model layer and the model layer:* A model relates to the arrangement and administration of the mesh. It is possible for a Bluetooth mesh gadget to be described, as a grouping of various models, which represents a part of the application and together they represent the device. An application is implemented above this stack.

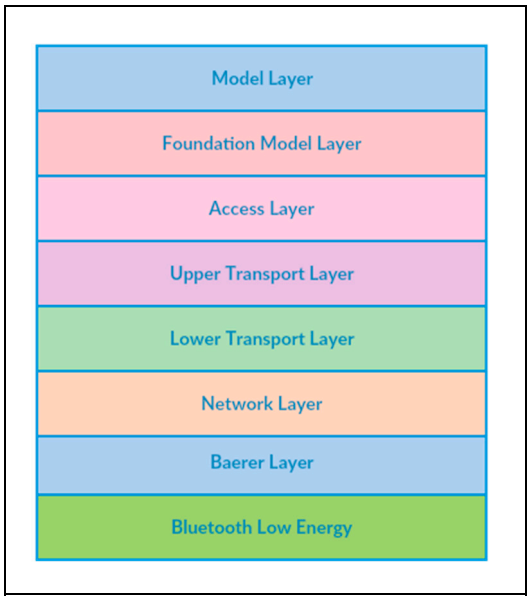

Figure 7. Bluetooth mesh stack [11].

2.3.2. Communication in a Bluetooth Mesh

Figure 8 depicts the communication between two Bluetooth mesh nodes. First, an event in the application layer of the transmitting node activates the need to transmit data to another node. It takes some processing time for data to be sent through the stack. Prior to sending data over the air, a random back-off mechanism is applied to hold data for some random interval between 0 milliseconds to at most $t_{maximum}$. When the back-off timer expires, the message is broadcasted. Data is announced on frequency channels 37, 38 and 39 in sequential order while a receptor node scans a channel one at a time.

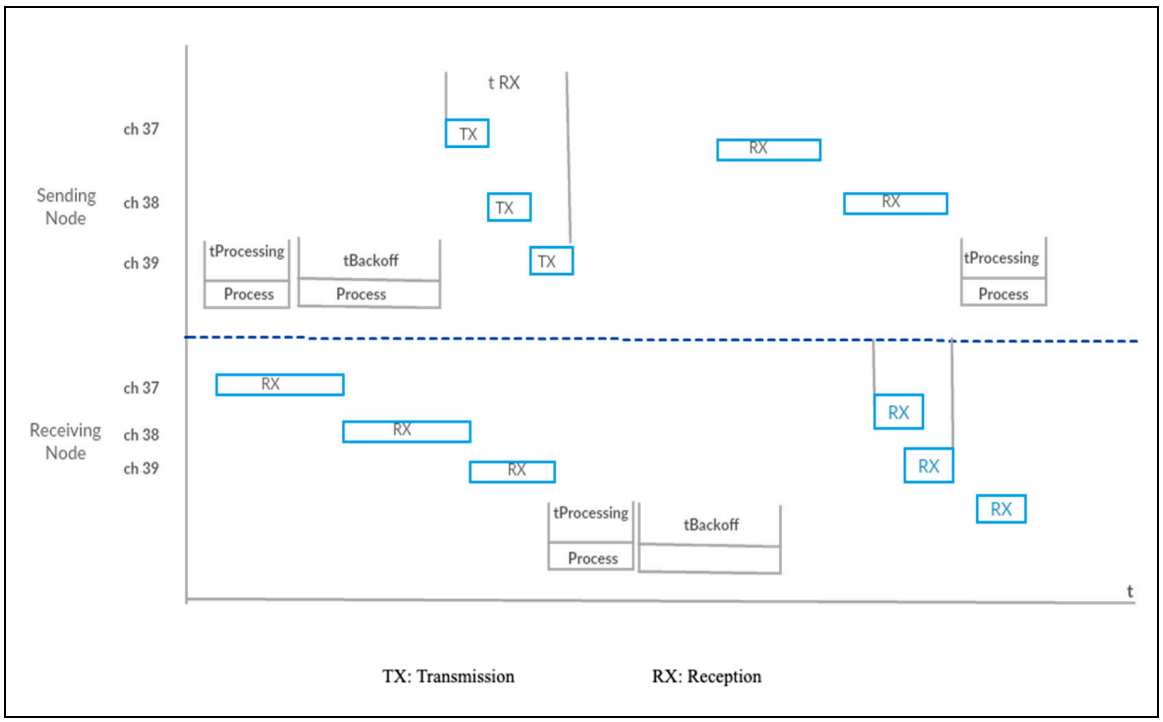

Figure 8. Communication in Bluetooth mesh [11].

Another communicating device may also scan for packets. The time taken for the packet to be received depends on the listening channel of the recipient device. Figure 8 shows how the receiving node receives the message on channel 39. When a message is received, it is passed up to the Application layer. After processing at the Application layer, an acknowledgement is then transmitted to the sender. Transmission of the acknowledgement follows the same mechanism discussed above. The data related to the acknowledgement has been received on channel 38, which implies that transmission time for channels 37 and 38 (as well as the channel switch time) must be considered [11].

2.3.3. Applications of Bluetooth Mesh

Bluetooth mesh networks are deployed in several industrial environments which implement IoT technologies. Bluetooth mesh uses include lighting control, asset tracking, environmental monitoring, beaconing for location services, and so on. The mesh network essentially simplifies the implementation of a multi-function network that can enable massive scaling without changes to the network infrastructure. For example, a single Bluetooth mesh network can support several services simultaneously.

Bluetooth mesh is also useful when low latency requirements exist. Silicon Labs benchmarks [12] demonstrate that Bluetooth mesh can provide a latency less than 10 milliseconds per hop with a single packet payload that is up to 11 bytes of data in size. Silicon Labs benchmarks also demonstrate Bluetooth mesh networks with a high number of nodes are capable of performing well. In a 240-node network test, with proper relay selection mechanism in place, 99% reliability was achieved by the mesh network. Furthermore, 98% of the packets arriving at the nodes had a latency lower than 60 milliseconds [13]. Table 2 presents a summary of the major improvements in each Bluetooth version.

Table 2. Improvements in Bluetooth versions [14].

Core Version	Issue Year	Major Improvements
1.0	1999	-
1.2	2003	Adaptive frequency hopping, inquiry-based RSSI
2.0	2004	2.1 Mbps peak data rates
2.1	2007	3.0 Mbps peak data rates
3.0	2009	24 Mbps peak data rates
4.0	2010	Lower energy consumption, broadcasting, lower connection latency
4.1	2013	Improved device power management by pairing that allows automatic powering up and down
4.2	2014	Improved security, low energy data packet length extension, link layer privacy
5.0	2016	Higher data rates (48 Mbps), better energy efficiency, higher broadcasting message capacity, larger range and strong point-to-point connection and reliability

3. Connecting Bluetooth Devices to IP-Based Networks

Bluetooth technology has evolved from being a wireless system used to connect PC peripherals into a very useful industrial and domestic IoT connectivity solution. In order to be used in an IoT scenario, Bluetooth needs to seamlessly integrate with IP-based networks. For example, in order to transfer Internet Protocol v6 (IPv6) packets over Bluetooth Low Energy (BLE), the BLE protocol stack needs to provide support for IPv6 in the form of an adaptation layer [15]. Next, we describe how a Bluetooth stack can offer support for IPv6 by using a typical Bluetooth networking scenario and commonly available devices. BlueZ [15], the widely used Bluetooth stack which provides support for core Bluetooth layers and protocols, can be used to implement a lightweight IPv6 stack. A BLE network consists of sensor nodes and routers. For instance, a Nokia N9, a Linux based smart phone integrated with TI WL1273 BLE chip is the router. Wireless key fobs with embedded TI CC2540 BLE chips are used as the sensor nodes. The lightweight IPv6 stack, ulPv6 [10] was originally integrated

with Contiki, an open source OS for microcontrollers, but can also be ported to TI CC2540 keyfob to support IPv6. The B1uetooth core system contains two major subsystems, the Controller and the Host. In order to keep the interoperability between different Bluetooth subsystems, the common layer called "Host Controller Interface (HCI)" is defined. In Linux, most of the host part can be implemented as kernel modules and the controller part is implemented in the chip provided by its vendor.

To implement the lightweight IPv6 protocol stack which avoids the modification of the existing IPv6 module of the Linux kernel, an adaptation layer is implemented below the IPv6 layer. Figure 9 shows the components of BlueZ. In the Logical Link Control and Adaptation Protocol (L2CAP) layer, the link type is set to BLE link and a new logical channel with channel ID Ox0007 is defined for transmitting IPv6 packets over BLE. Furthermore, to meet the requirements of application layer and IPv6's Maximum Transmission Unit (MTU), the Segmentation and Reassembly (SAR) functionality is implemented in this layer by using L2CAP Information Frame (I-frame) defined in Bluetooth Core Specification [2]. The 6LoWPAN adaptation layer is used to handle the data flow between the transport layer and the L2CAP layer, and at the same time provides a network interface for upper layer applications. Both, the IPv6 Header Compression and context management functionality are implemented in this adaptation layer.

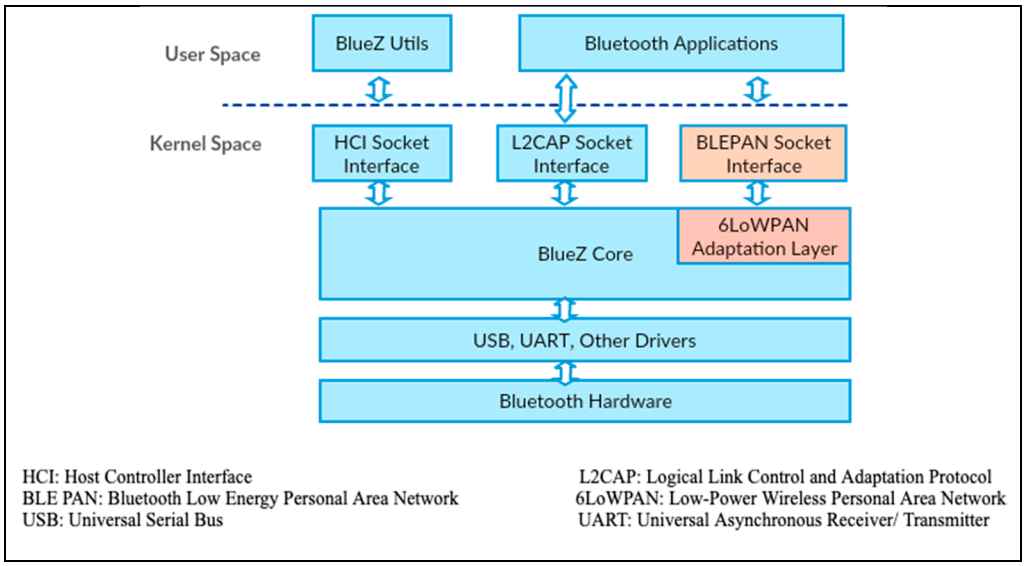

Figure 9. BlueZ stack [15].

In Wireless Sensor Networks (WSNs), a node equipped with multiple sensors may bind with one or more IPv6 addresses that provide services. Working with a web service provider, the vendor can program the IPv6 address for the predefined remote server into the sensors. Alternatively, one known server can provide DNS-like services to distribute global IPv6 addresses for various web services. In each case, after the connection has been established between the sensor and the router, the sensor may need to register one or more remote server's IPv6 addresses that it will used to communicate with by sending the context information to the router. Next, the router responds with the corresponding Context IDs (ClDs) for contexts it has received in its response. After the context information is exchanged successfully, the sensor is able to communicate with the remote server through the IPv6 router. The procedure to update the context information is similar.

In addition to allowing the context exchange procedure initiated by the sensor, the context exchange mechanism can also include the capability of compression. According to RFC 6282 [16] the fields which cannot be compressed should be carried in-line following the compressed header. If not frequently changed, these fields can be stored in the context instead of being carried in-line. The extension of context information will be helpful in optimizing the header compression for some specific use cases. For example, long strings can be used as the payload if a configuration file formatted

in JavaScript Object Notation (JSON) or eXtensible Markup Language (XML) is transmitted over BLE, which can also be stored in the context to compress the content of payload [15].

4. Communication between Bluetooth and Zigbee

4.1. Zigbee

Zigbee [17] is also a low power network technology like Bluetooth Low Energy. It is also being used for several purposes such as monitoring, sensing, and so on where a few data packets are sent over short distances and the application requires very low power consumption. Zigbee supports different network topologies such as Star, Mesh, and Tree, and is also used in local area sensor data networks on 2.4 GHz Industrial, Scientific, Medical (ISM) frequency band to transmit the data at 250 Kbps [18].

4.2. Bluetooth-Zigbee Interoperability

Interoperability is important for the large-scale adoption of the IoT [18]. To enable interoperability between Bluetooth and Zigbee, a gateway device should be implemented that converts between Bluetooth and Zigbee data formats. An example of such a gateway is the BlueBee system [19]. Bluebee is a gateway that translates data from Zigbee devices to Bluetooth data. The BlueBee node contains two main modules to manage the system activities: Data Communications Equipment (DCE) Module and Digital Terminal Equipment (DTE) Module (as shown in Figure 10).

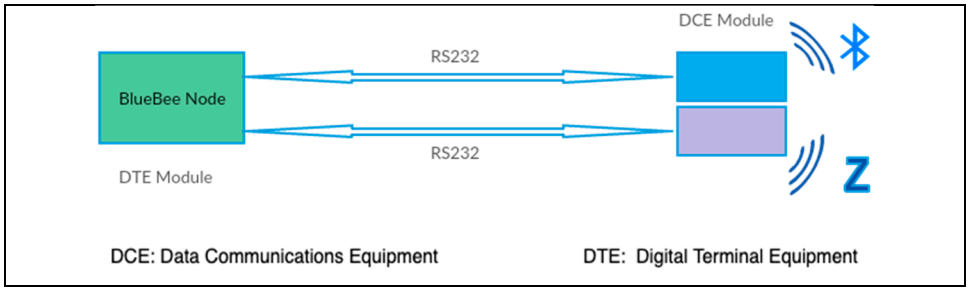

Figure 10. BlueBee: Bluetooth-Zigbee gateway [19]. Reproduced with permission from Garcia et al., IEEE 2011.

THE DTE module is used to configure the node and show data transmitted between nodes. The DTE module includes specific modules as follows:

- *dsPIC Microcontroller*: Node settings are implemented via ATtention (AT) commands transmitted to the DCE module.
- *Graphic Liquid Crystal Display (GLCD):* GCLD allows the user to see various system settings and messages.
- *Keyboard*: Allows the user to type in text values.
- *RS232 transceiver*: Changes the RS232 voltage levels to UART voltages.
- *Reset button*: Helps restart the DTE module.

The DCE module includes the following components:

- *Zigbee module*: It is an Original Equipment Manufacturer (OEM) serial port adapter OZS311i Zigbee/IEEE 802.15.4 module launched by ConnectBlue [20], a manufacturer for data products and solutions. The OZS311i is an IEEE 802.15.4 implementation, with an internal antenna of approximately three-hundred-meter range.
- *Bluetooth module*: This is the OEM serial port adapter OBS433i that is a long range Bluetooth 2.1+EDR module (class 1), with support for the Serial Port Profile (SPP) for fast and secure transparent serial data transmissions, and the Personal Area Networking Profile (PAN). It is worth

noting that the Serial Port Profile specifies the utilization of the Radio Frequency COMMunication (RFCOMM) protocol to emulate RS232 cable communications [21].

A Bluetooth module has different operating modes (as discussed below), which determines BlueBee's connectivity as well as discovery of other BlueBee nodes. SPP defines serial connections. The BlueBee node can work in discoverable or non-discoverable modes. The Zigbee mode can allow or disallow Zigbee in the BlueBee node. The Zigbee Mode allows works in both discoverable and non-discoverable modes.

The Zigbee Channel Mode connects with other Zigbee devices utilizing the radio channel and implements authentication using Zigbee security mode. The BlueBee node can manage both Bluetooth and Zigbee, and convert Zigbee data into Bluetooth data (in the BlueBee Data Mode), but not vice versa. The BlueBee node transmits data using Bluetooth and Zigbee. For a successful transmission, a user must be active and have Bluetooth and Zigbee modules configured. This resets the operation mode of the BlueBee node to data mode for a successful connection with the respective devices [19]. The combination of Bluetooth and Zigbee WPAN technologies can be used to deploy sensor networks that can be utilized in Industrial IoT (IIoT) [22,23] for location services [24,25], as well as for capturing various types of environmental data [24,26]. Both solutions provide low-cost and low energy consumption solutions for designing embedded systems.

5. Bluetooth Security

There are billions of Bluetooth devices in use today. These devices are exposed to different types of threats. Bluetooth security solutions need to constantly evolve to mitigate emerging threats. Similar to any other wireless communication systems, Bluetooth transmissions can be deliberately jammed or intercepted. False or modified information can be passed to the devices by malicious users. Security threats in Bluetooth can be divided into three major categories as follows [27]:

- *Disclosure threat*: The information can leak from the target system to an eavesdropper that is not authorized to access the information.
- *Integrity threat*: The information can be deliberately altered to mislead the recipient.
- *Denial of Service (DoS) threat*: The users can be blocked from gaining access to a service by making it either unavailable or severely limiting its availability to an authorized user.

Bluetooth devices are exposed to malicious intervention during the process of pairing with another device. These weaknesses are primarily due to flaws in the link key establishment protocol [27] which is required for devices to pair (i.e., establish a trusted relationship), and because session encryption is not mandatory. Therefore, attacks can occur prior to the completion of pairing. Link keys can still be sniffed after the pairing completes and used to perform illegal authentication or Man-in-the-Middle (MITM) attacks [27]. We present a summary of Bluetooth attacks below.

- PIN cracking

For data exchange between Bluetooth devices, trust has to be established between them. This process is known as pairing and is implemented by exchanging secret codes, often referred to as the Personal Identification Number (PIN). PIN length can be up to 8 bytes. Pairing is implemented in the following steps: Initialization, Link key Generation, Authentication, and Encryption. An attacker can eavesdrop the entire process of pairing and authentication and collect all the messages. The attacker can then use a brute force algorithm to identify the PIN used. If the device's Media Access Control (MAC) address is known, then by using a 128-bit guessed number, an accurate initialization key could be identified. The next step is to find the shared session link key by using all the collected data. Provided the data gathered is accurate, the PIN is determined without much effort. After cracking the pin, the attacker can pair with the target device and access information illegally [28].

Recently proposed security solutions that can mitigate intrusions related to pairing [27,29] include the use of a combination of long public/private keys, which are harder to crack.

- MAC spoofing

Spoofing is done prior to encryption and when the piconet forms. Devices can authenticate each other by producing link-keys. While the attack is ongoing, attackers could impersonate an alternative client. There is a possibility for attackers to end connections or modify data during transmission utilizing certain hacking tools [29].

- Man-In-The-Middle (MITM) Attack

The first MITM intrusion (Figure 11) was developed on the idea that the hackers are aware of the shared key used by the Bluetooth devices. It is also possible to get the link key by other methods such as eavesdropping and brute-forcing the PIN. A hack that utilizes manipulating the Bluetooth clock, involves devices that require the same hopping sequence on different clocks. A hack can be achieved by replying to the page request of the master device ahead of the slave. This restarts the paging with the slave using a different clock. MITM attacks can be launched during Secure Simple Pairing (SSP). One type of attack during the SSP process focuses on wrong data transmitted during the initial steps of SSP called the Input/ Output (IO) phase in which devices that are interested in pairing perform an exchange of IO capabilities [30]. Another type of SSP attack needs a visual contact with the victim's Bluetooth devices (such as direct line-of-sight or possibly a video camera that is hidden) to guarantee that a lower security association model choice is made by the Bluetooth device user. Once the attacker (MITM) has visual access to the victim's devices, the attacker acts before the legitimate user to establish Bluetooth connections to both victims' devices and to initiate the IO phase in which the less secure association model can be forcefully selected [31].

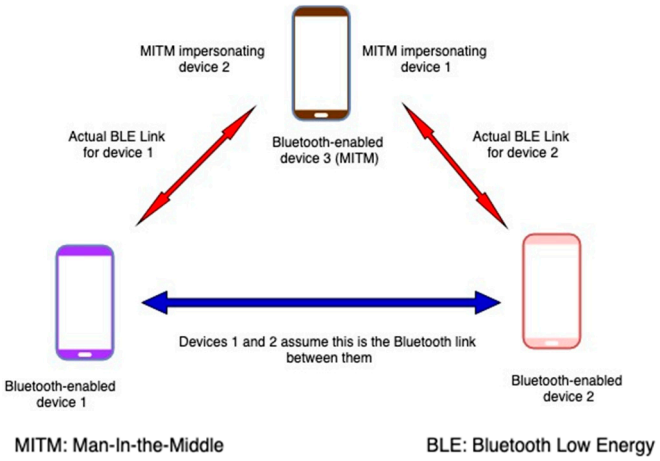

Figure 11. Man-in-the-Middle (MITM) attack on Bluetooth.

To mitigate MITM attacks, we need to incorporate more specific piconet information into the pairing process. For example, timestamps and nested mutual authentication can be used to determine the legitimacy of a device's challenge before responses are sent [27].

- Bluejacking attack

Bluejacking makes use of the Bluetooth technology to send unsolicited/unwelcomed messages to Bluetooth enabled devices. The receiver does not know the sender of the message. It only receives the message along with the name and model of the sender's phone. Bluejacking is instigated by an attacker (also known as bluejacker or bluejack addict) who forwards unsolicited messages to a Bluetooth-enabled device user. When the connection is established, the bluejacker tries to send a message to the recipient. The actual message sent to the user's device does not cause any harm but is used to cause the user to counter react in some manner or add a new contact to the device's address book [32].

Devices that are set in non-discoverable, hidden or invisible mode are not susceptible to blue jacking [32].

- BlueSnarfing attack

BlueSnarfing allows unapproved access to a Bluetooth node. In this attack, the attacker hacks the node in order to access its contact book, document files, etc. It could also possibly forward messages and calls to another device [28].

BlueSnarfing can be avoided by deactivating discovery mode of the device, keeping the device in invisible mode, and by utilizing tools that restrict the device connection to known devices only.

- Bluebugging attack

Bluebugging is perhaps an attack of most concern. In a bluebugging attack, the intruder gets illegal access to a device and can then run commands or implement other actions such as making phone calls, and so on. Such actions can result in major problems. Bluebugging exploits a security flaw in the firmware of some older Bluetooth devices (mostly those using Bluetooth classic) to gain access to the device and its commands [33].

Bluebugging can be avoided by switching off the Bluetooth radio capability while not in use since Bluebuggers can only make a connection when Bluetooth is enabled. It is also useful to scan all incoming multimedia messages for viruses. Bluebuggers often gain access to the device by transmitting such information to it [34].

- Bluesmack attack

A Bluesmack attack is the Bluetooth equivalent of the Ping-of-Death denial-of-service attack. This is a buffer overflow issue which uses L2CAP echo messages, comprising a large volume of packets sent to the victim node in a short interval of time [35].

- BluePrinting attack

Blueprinting is a method to remotely extract information from Bluetooth-enabled devices. Blueprinting can be used for generating statistics about manufacturers and models and to find out whether there are devices in range that have weak Bluetooth security [36].

Safety procedures include switching off Bluetooth functionality whenever not in use, employing authentication and encryption whenever possible, and never pairing with an unknown device.

- Backdoor attack

The backdoor attack involves establishing a trust relationship through the "pairing" mechanism but ensuring that it no longer appears in the target's register of paired devices. In this way, unless the owner is actually observing their devices at the precise moment a connection is established, they are unlikely to notice anything unusual, and the attacker may be free to continue to use any resource that a trusted relationship with that device grants access to. This means that not only can data be retrieved from the phone, but other services, such as modems, or Internet, Wireless Application Protocol (WAP) and General Packet Radio Service (GPRS) gateways could be used [37].

The Backdoor attack can only be implemented if the target device's BT_ADDR is known. Therefore, this attack can be mitigated by hiding the Bluetooth devices BT_ADDR (for example by enabling encryption while establishing Bluetooth connections) [38].

- DoS attack

In a Denial-of-Service (DoS) attack, the attacker attempts to prevent valid users from accessing the service by sending a large number of messages to the Bluetooth device. Denial of service attacks may aim to drain the battery life of the Bluetooth device through constant activity. For example, an attacker can send repeated pairing requests or device information requests to a Bluetooth device. This constant activity quickly drains the device battery and result in a Battery draining DoS attack [39].

- BD_ADDR attack

The attack occurs when a 'bug' is kept within coverage area of a Bluetooth gadget. The bug duplicates the BD_ADDR of the target device. It is worth pointing out that the Bluetooth Device Address (or BD_ADDR) is a unique 48-bit identifier assigned to each Bluetooth device by the manufacturer. Whenever a Bluetooth node attempts to connect with the target device, both, the target device and the bug respond simultaneously and cause jamming. This allows a denial of access for the actual legitimate user.

- SCO/eSCO attack

This attack is based on a real-time, two-way voice packet. It acquires a lot of a Bluetooth piconet's attention so that the genuine piconet devices cannot access to the service in an acceptable time frame. Establishing a SCO or an enhanced-SCO (e-SCO link) with the piconet master can easily lead to this attack.

- L2CAP guaranteed service attack

The intruder asks for maximum bandwidth and lowest latency performance. This leads to rejection of all other requests since bandwidth is now entirely reserved for the intruder. [40].

- Fuzzing

This intrusion involves sending malformed or otherwise non-standard data to a device's Bluetooth radio and observing how the device reacts. When a device's response is slowed or stopped by these attacks, this indicates that a serious vulnerability potentially exists in the protocol stack [41].

- BlueBorne

This attack allows an attacker to exploit some insecure implementations of Bluetooth on most platforms (Linux devices, Android devices, Amazon and Google home devices) to control or extract information remotely [42].

- MultiBlue

In this attack, an intruder can access the node to be hacked. The MultiBlue dongle, a Bluetooth capable 4 GB thumb drive, is utilized to take control of the target device. The intruder utilizes the MultiBlue application to send pairing requests to discoverable nodes. The targeted device then presents a code (a pre-shared key) which the MultiBlue application uses as the authentication key. The attacker then has full node control [29].

- Cabir worm

The Cabir worm is a malicious software that uses Bluetooth technology to search for available Bluetooth devices and send itself to them. The user has to manually accept the worm and install the malware in order to infect the phone. The Mabir worm is essentially a variant of the Cabir worm which uses Bluetooth and Multimedia Messaging Service messages (MMS) to replicate.

- Helemoto

This attack is like the Bluebugging attack but exploits the poor implementations of a "trusted device" management on certain phones. As with Bluebugging attacks, the attacker pretends to send a Virtual Contact File (vCard) to an unverified Bluetooth Object Exchange (OBEX) Push Profile on the victim's device. The OBEX is a profile in the Bluetooth specification that enables a Bluetooth device to send and receive an object (file) with another Bluetooth device.

Once the attack begins, the attacker disrupts the transfer process and the victim lists the attacker's phone as a trusted device. The attacker then associates with the victim's phone and issues AT commands [35].

- Reflection/Relay

An intruder need not be aware of any confidential data as only relays sensitive data from one node to the other in the authentication phase. [41]. Table 3 presents a summary of Bluetooth attacks that occur before and after the pairing process.

Table 3. List of Bluetooth attacks.

Attacks Prior to Pairing	Attacks after Pairing
BlueJacking	Backdoor
BlueSnarfing	Denial of Service (DoS)
BlueBugging	Worm
MAC spoofing	Bluesmack
Helemoto	MultiBlue
BluePrinting	Offline PIN recovery
Fuzzing	BD_ADDR
BlueBorne	Reflection/Relay

Security Risks Specific to Bluetooth Low Energy

The main security threats related to the process of pairing and to BLE as such are passive identity tracking, MITM, and eavesdropping.

Inactive snooping occurs when a third device eavesdrops information transmitted amongst paired devices. BLE mitigates the threat by implementing various encryption types such as Advanced Encryption Standard (AES)-Counter with Cipher Block Chaining Message Authentication Code (CBC-MAC) (CCM) cryptography, etc.

MITM attacks in BLE occur when a malicious device, mimics other legitimate devices and pairs with them. In this scenario, the BLE Generic Access Profile (GAP) connects to the malicious device which in turn routes the communication between the two other devices. Legitimate devices believe they are connected to each other while they are actually connected to the malicious user's device. This enables the intruding device to capture other users' data and also alter and re-insert new data in a transmission.

Identity tracking occurs when a mischievous user associates the address of a BLE device with a certain user and then tracks that user. BLE mitigates this threat by intermittently altering the address of the device [43].

6. Recent Research Results on Bluetooth

In this section, we briefly review some of the recent research efforts that have been undertaken on Bluetooth technology in various application domains over the last few years.

6.1. Health

Many research works have focused on demonstrating the use of Bluetooth Low Energy (BLE) for developing wearable IoT sensor networks for health applications [44–48]. These BLE health sensor networks implement different types of monitoring including temperature, humidity, and carbon dioxide for environmental observations or body temperature and heart rate for physiological conditions [44,46]. A BLE sensor network consisting of emotion sensors has been demonstrated to help observe a body's emotion regulation process. These can serve as wearable biofeedback devices and are useful in clinical and psychological research [45].

Remote health monitoring using wearable sensors for continuous examination of blood pressure, heart rate, and body temperature is also becoming a popular Bluetooth use-case [46]. The vital data can be communicated from sensors via Bluetooth to nearby "gateway" devices such as Android smart phones, and so on. It has been demonstrated that Bluetooth provides better transmission rate than Zigbee and better cost-efficiency than the Global System for Mobile Communication (GSM) [46].

Recently, researchers have also investigated Bluetooth for developing pervasive healthcare solutions for mental health [47]. By utilizing a Bluetooth-based wireless monitoring system, researchers have shown the possibility of monitoring mental health in individuals by collecting and analyzing physiological signals such as short-term Heart Rate Variability (HRV) from a wristband worn during natural daily activities. Remote management of patients for chronic diseases in general has also been the focus of research lately. The remote monitoring kits include wireless sensors with Bluetooth Low Energy connectivity to a local gateway (tablet/smartphone with specialized software) and a network connection to a centralized cloud-oriented application for data processing and storage [48].

6.2. Location Tracking

The research community is also currently exploring the usability of BLE for *location tracking and indoor positioning systems* [49–53]. For example, BLE proximity beacons can be utilized for determining the location of buses without requiring the usage of any Global Positioning System (GPS) devices. BLE beacons are being deployed on buses so that they can be tracked, and by installing BLE detection devices at selected bus stops along the route, bus arrivals can be detected. The authors of [49] showed that the detection of BLE beacons is very accurate and tracking bus locations without using a GPS device is a cost effective method. BLE beacon technology is also being explored for position tracking and for identifying people in micro-locations such as buildings [50]. Indoor navigation systems with BLE technology use Bluetooth beacons that emit radio frequency signals.

One issue with BLE-based indoor positioning systems is the fluctuation in the Received Signal Strength (RSS) because of fading. Recent research [51] has proposed the use of spatial diversity and frequency diversity to handle the fluctuations and provide better positioning accuracy. There is also a lot of focus on developing new fingerprinting-based algorithms to address issues of accuracy, precision, and time complexity when implementing positioning using BLE-beacons [52]. Efforts are being focused on improving the time-complexities of these algorithms [52] as well as employing machine learning algorithms together with BLE to improve location accuracy [53]. Some BLE systems have shown an improvement in the positioning accuracy by about 15% compared to other positioning alternatives such as GPS, and so on [54].

6.3. Security

As the number of Bluetooth devices keeps increasing, there is expanding growing research focus on analyzing the *security* of BLE devices. Sevier et al. [55] have shown that Bluetooth Low Energy's most severe vulnerability is the *temporary key* used in the pairing process. The pin is 6-digits long and therefore too small to prevent a modern computer from generating that key using the brute-force approach. Therefore, recent research recommends that Bluetooth Low Energy should not yet be used in mission critical systems or those with sensitive data.

Studies on BLE attacks [56] have suggested the development of a protocol that provides an extra layer of security before the generation of the temporary key, and to generate a large and more random key which is difficult to determine easily via brute force methods. Apple's Bluetooth Low Energy (BLE) Continuity protocol [57] was designed to support interoperability and communication between iOS and macOS devices. Although the protocol provides a seamless communication experience, it leaks device and behavioral data to nearby listeners. A small amount of information is leaked at a time. However, this can be used to identify and track devices over long periods of time. BLE technology has privacy challenges that need to be addressed before its secure deployment becomes possible.

6.4. Smart Homes

Internet of Things (IoT) is having a huge impact on our daily lives. Several home gadgets now have the capability to connect to the Internet. BLE technology is playing a significant role in enabling the deployment of *smart homes*.

Recent research efforts [58] have proposed an energy management approach for smart homes. Bluetooth Low Energy is utilized for communication among home appliances. The approach helps in reducing the peak load demand and electricity consumption charges and reaps huge savings. Researchers have been investigating the use of BLE with artificial neural network support for home energy management [59]. This approach helps to predict the home energy requirements at different times of the day throughout the week and helps to implement *smart grid* applications.

Other smart home applications [60] using BLE include home lighting, smart door locking, cloud monitoring, and others. Cloud applications provide remote monitoring and control of IoT devices. The applications allow the user to view heating history and make adjustments remotely to the temperature settings. The cloud application can read the temperature service from the sensors and the thermostat at all times but can only set the temperature when a Short Message Service (SMS) message confirms that user has instructed it to do so.

One recent study [61] has also investigated how a Bluetooth mesh can be used for deploying smart building applications. The Bluetooth mesh permits gadgets, such as smart phones, having a 4.0 or higher BLE version, with GATT support to join the mesh network. The mesh network is used to deploy a smart doorbell application, which has many nodes connected in a mesh network to send messages amongst nodes. Devices located at building doors represent a doorbell and send messages to a central node, also referred to as an Internet gateway. The range of the network is extended via relay nodes. A message is generated by pressing a button on the node at the door. This message signals to the staff regarding incoming visitors. Upon receiving the message, the gateway transmits it to all network devices as a notification to the staff. The building's space and structure need to be taken in to account when designing a mesh network [60].

7. Conclusions

Bluetooth has been around for over two decades and during this time, the technology has significantly evolved and the Bluetooth market has continued to expand at a fast rate. The development and deployment of evolving Bluetooth technologies have resulted in various Bluetooth versions namely, 1.2, 2.0, 2.1, 3.0, 4.0, 4.1, 4.2, and 5.0. Each version has brought about many improvements and benefits over its predecessor to its users and has paved the way for new use cases. Version 4.0 introduced Bluetooth Low Energy (BLE) or Bluetooth Smart. With the advent of the Internet of Things (IoT), manufacturers are increasingly interested to find ways to use short-range wireless connectivity for numerous types of battery powered devices. Bluetooth 5 [2] aims at addressing the requirements for IoT applications. Higher range and transmission speed are being made possible through multiple enhancements to the BLE radio PHY, as well as an increase of maximum transmit power from +10 to +20 dBm. One new radio PHY (Bluetooth version 4.0) [2] is expected to support a higher range but at the expense of a lower data transfer speed. Another new PHY (Bluetooth version 5.0) doubles the transmission speed (about 2 Mbps) which enables more efficient transfer of larger amounts of data. The high speed PHY can also support audio over BLE (like the current classic Bluetooth).

In this work, we have presented an overview of the Bluetooth technology together with some current and future trends in Bluetooth device designs. We have described the three main types of Bluetooth configurations. We discussed the interworking of Bluetooth with other WPAN technologies such as Zigbee along with Bluetooth internetworking with IP-based networks. We discussed major cyberattacks that can be launched on Bluetooth devices and solutions to mitigate such attacks. Finally, we discussed some of the latest research efforts on Bluetooth technology in areas such as health, location tracking, security and smart homes.

Author Contributions: S.Z.: conceptualization; writing original draft preparation; writing—review and editing; supervision; F.S.: writing original draft preparation; writing—review and editing; Z.B.: writing—review and editing.

Funding: This research received no external funding.

Acknowledgments: We thank the anonymous reviewers for their valuable comments which helped us improve the quality, content, and organization of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

ACL	Asynchronous Connectionless link
ADV	Advertising
API	Application Programming Interface
AT	Attention
BLE	Bluetooth Low Energy
BR	Basic Rate
CID	Context ID
CRC	Cyclic Redundancy Checks
DCE	Data Communications Equipment
DoS	Denial of Service
DPSK	Differential Phase Shift Keying
DQPSK	Differential Quadrature Phase Shift Keying
DTE	Digital Terminal Equipment
EDR	Enhanced Data Rate
FM	Frequency Modulation
GAP	Generic Access Profile
GATT	Generic Attribute Profile
GLCD	Graphic Liquid Crystal Display
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communication
HCI	Host Controller Interface
HS	High Speed
IC	Integrated Circuit
IIoT	Industrial IoT
IoT	Internet of Things
IP	Internet Protocol
ISM	Industrial, Science and Medical
JSON	JavaScript Object Notation
L2CAP	Logical Link Control and Adaptation Layer Protocol
LAN	Local Area Network
LE	Low Power
LMP	Link Management Protocol
LTE	Long Term Evolution
MAC	Media Access Control
MITM	Man-In-The-Middle
OBEX	Bluetooth Object Exchange
OEM	Original Equipment Manufacturer
PC	Personal Computer
PDU	Protocol Data Unit
PHY	Physical
POI	Point of Interest
PSK	Frequency Modulation
RF	Radio Frequency
RFCOMM	Radio Frequency COMMunication
RX	Reception
SAR	Segmentation and Reassembly
SCO	Synchronous Connection-Oriented
SDP	Service Discovery Protocol
SIG	Special Interest Group
SM	Security Manager
SMS	Short Message Service
SPP	Serial Port Profile

SSP	Secure Simple Pairing
TCP	Transmission Control Protocol
TTL	Time-To-Live
TX	Transmission
UDP	User Datagram Protocol
vCard	Virtual Contact File
WAN	Wide Area Network
WAP	Wireless Application Protocol
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Network
XML	eXtensible Markup Language

References

1. Sairam, K.; Gunasekaran, N.; Redd, S. Bluetooth in wireless communication. *IEEE Communications Magazine*, 7 August 2002; 90–96.
2. Bluetooth. Available online: https://www.bluetooth.com/ (accessed on 18 July 2019).
3. Torvmark, K. Three Flavors of Bluetooth®: Which One to Choose? *Texas Instruments White Paper*. 2014. Available online: http://www.ti.com/lit/wp/swry007/swry007.pdf (accessed on 7 September 2019).
4. Jaycon Systems Bluetooth Technology: What Has Changed Over the Years. Available online: https://medium.com/jaycon-systems/bluetooth-technology-what-has-changed-over-the-years-385da7ec7154 (accessed on 7 September 2019).
5. Bluetooth. Available online: https://gtrusted.com/product/121524 (accessed on 7 September 2019).
6. Mikulka, J.; Hanus, S. Bluetooth EDR Physical Layer Modeling. In Proceedings of the 18th International Conference on RadioElectronics, Prague, Czech Republic, 24–25 April 2008.
7. Chada, S.; Singh, M.; Pardeshi, S. Bluetooth Technology: Principle, Applications and Current Status. *Int. J. Comput. Sci. Commun.* **2013**, *4*, 16–30.
8. Dian, F.; Yousefi, A.; Lim, S. A practical study on Bluetooth Low Energy (BLE) Throughput. In Proceedings of the IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 1–3 November 2018.
9. Ethical Aspects of Nano Technology and Neuro-Computing. Bionic Gate. Available online: http://www.bionicgate.com/the-internet-of-things/ (accessed on 7 September 2019).
10. Newman, N. Apple iBeacon technology briefing. *J. Direct Data Digit. Mark. Pract.* **2014**, *15*, 222–225. [CrossRef]
11. Baert, M.; Rossey, J.; Shahid, A.; Hoebeke, J. The Bluetooth Mesh Standard: An Overview and Experimental Evaluation. *Sensors* **2018**, *18*, 2409. [CrossRef] [PubMed]
12. The Highlights of Bluetooth Mesh Networking Technology. Silicon Labs. Available online: https://www.eetimes.com/document.asp?doc_id=1333715# (accessed on 7 September 2019).
13. AN1137: Bluetooth Mesh Network Performance. Silicon Labs. Available online: https://www.silabs.com/documents/login/application-notes/an1137-bluetooth-mesh-network-performance.pdf (accessed on 7 September 2019).
14. Positioning in Bluetooth and UWB Networks—Scientific Figure on ResearchGate. Available online: https://www.researchgate.net/figure/Milestones-in-the-Bluetooth-Evolution-Path_tbl2_316849796 (accessed on 7 September 2019).
15. Wang, H.; Xi, M.; Liu, J.; Chen, C. Transmitting IPv6 packets over Bluetooth Low Energy based on BlueZ. In Proceedings of the 15th IEEE International Conference on Advanced Communications Technology, PyeongChang, Korea, 27–30 January 2013.
16. Ed, J.; Thubert, P. *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*; RFC 6282; IETF: Fremont, CA, USA, 2011.
17. Ramya, M.; Shanmugaraj, M.; Prabakaran, R. Study on Zigbee Technology. In Proceedings of the 3rd IEEE International Conference on Electronics Computer Technology, Kanyakumari, India, 8–10 April 2011.
18. Rahman, T.; Chakraborty, S. Provisioning Technical Interoperability within Zigbee and BLE in IoT Environment. In Proceedings of the 2nd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), Kolkata, India, 4–5 May 2018.

19. Cano, E.; Garcia, I. Design and Development of a BlueBee Gateway for Bluetooth and Zigbee Wireless Protocols. In Proceedings of the IEEE Electronics, Robotics and Automotive Mechanics Conference, Cuernavaca, Mexico, 15–18 November 2011.
20. ConnectBlue. Quality Data Products and Accessories. Available online: http://www.connectblue.com.au/ (accessed on 7 September 2019).
21. Moron, M.; Luque, R.; Casilari, E. Modeling of the transmission delay in bluetooth piconets under serial port profile. *IEEE Trans. Consum. Electron.* **2010**, *56*, 2080–2085. [CrossRef]
22. Chen, F.; Wang, N.; German, R.; Dressler, F. Simulation study of IEEE 802.15.4 LR-WPAN for industrial applications. *Wirel. Commun. Mob. Comput.* **2010**, *10*, 609–621. [CrossRef]
23. Heynicke, R.; Krush, D.; Cammin, C.; Scholl, G.; Kaercher, B.; Ritter, J.; Gaggero, P.; Rentschler, M. O-Link Wireless enhanced factory automation communication for Industry 4.0 applications. *J. Sens. Sens. Syst.* **2018**, *7*, 131–142. [CrossRef]
24. Ros, M.; D'Souza, M.; Postula, A.; MacColl, I. Location based services with personal area network for community and tourism applications. In Proceedings of the IET International Communication Conference on Wireless Mobile and Computing, Shanghai, China, 14–16 November 2011.
25. Miyagawa, Y.; Segawa, N. Construction of Indoor Location Search System Using Bluetooth Low Energy. In Proceedings of the IEEE Nicograph International, Kyoto, Japan, 2–3 June 2017.
26. Pule, M.; Yahya, A.; Chuma, J. Wireless sensor networks: A survey on monitoring water quality. *J. Appl. Res. Technol.* **2017**, *15*, 562–570. [CrossRef]
27. Minar, N.; Tarique, M. Bluetooth Security Threats and Solutions. *Int. J. Distrib. Parallel Syst.* **2012**, *3*, 127. [CrossRef]
28. Hassan, A.; Bibon, S.; Hossain, M.; Atiquzzaman, M. Security threats in Bluetooth technology. *Comput. Secur.* **2018**, *74*, 308–322. [CrossRef]
29. Lonzetta, A.; Cope, P.; Campbell, J.; Mohd, B. Security Vulnerabilities in Bluetooth Technology as used in IoT. *J. Sens. Actuator Netw.* **2018**, *7*, 28. [CrossRef]
30. Toivanen, P.; Haataja, K. Two practical man-in-the-middle attacks on Bluetooth secure simple pairing and countermeasures. *IEEE Trans. Wirel. Commun.* **2010**, *9*, 384–392.
31. Sandhya, S.; Devi, S. Contention for Man-in-the-Middle Attacks in Bluetooth Networks. In Proceedings of the Fourth IEEE International Conference on Computational Intelligence and Communication Networks, Mathura, India, 3–5 November 2012.
32. Kaviarasu, S.; Mathupandian, P. Bluejacking Technology: A Review. *Int. J. Trend Res. Dev.* **2016**, *3*. [CrossRef]
33. Padgetta, J.; Batra, M.; Holtmann, M.; Chen, L.; Scarfone, L. *Guide to Bluetooth Security*; NIST Special Publication: Gaithersburg, MD, USA, 2017; Volume 800, p. 121.
34. Dhuri, S. Bluetooth Attack and Security. *Int. J. Curr. Trends Eng. Res.* **2017**, *3*, 76–81.
35. Browing, D.; Kessler, G. Bluetooth Hacking: A Case Study. Available online: https://www.garykessler.net/library/bluetooth_hacking_browning_kessler.pdf (accessed on 7 September 2019).
36. Trifinite: BluePrinting. Available online: https://trifinite.org/trifinite_stuff_blueprinting.html (accessed on 7 September 2019).
37. Musale, V.; Apte, S. Security Risks in Bluetooth Devices. *Int. J. Comput. Appl.* **2012**, *51*.
38. Panse, P.; Panse, T. A Survey on Security Threats and Vulnerability Attacks on Bluetooth Communication. *Int. J. Comput. Sci. Inf. Technol.* **2013**, *4*, 741–746.
39. Satam, P.; Satam, S.; Hariri, S. Bluetooth Intrusion Detection System. In Proceedings of the 15th IEEE International Conference on Computer Systems and Applications, Aqaba, Jordan, 28 October–1 November 2018.
40. Dubey, V.; Vaishali, K.; Behar, N.; Vishwavidyalaya, G. A Review on Bluetooth Security Vulnerabilities and a Proposed Prototype Model for Enhancing Security against MITM Attack. *Int. J. Res. Stud. Comput. Sci. Eng.* **2015**, 69–75.
41. Tsira, V.; Nandi, G. Bluetooth Technology: Security Issues and its Prevention. *Int. J. Comput. Technol. Appl.* **2014**, *5*, 1833–1837.
42. BlueBorne. Available online: https://armis.com/blueborne/ (accessed on 18 July 2019).
43. Bon, M. A Basic Introduction to BLE Security. Available online: https://www.digikey.com/eewiki/display/Wireless/A+Basic+Introduction+to+BLE+Security (accessed on 7 September 2019).
44. Wu, F.; Wu, T.; Yuce, M. Design and Implementation of a Wearable Sensor Network System for IoT-Connected Safety and Health Applications. In Proceedings of the IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 15–18 April 2019.

45. Lam, S.; Szypula, A. Wearable emotion sensor on flexible substrate for mobile health applications. In Proceedings of the IEEE Sensors Applications Symposium (SAS), Seoul, Korea, 12–14 March 2018.
46. Majumdar, S.; Rahman, M.; Islam, M.; Ghosh, D. Design and Implementation of a Wireless Health Monitoring System for Remotely Located Patients. In Proceedings of the 4th IEEE International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT), Dhaka, Bangladesh, 13–15 September 2018.
47. Park, J.; Kim, J.; Kim, S. A Study on the Development of a Day-to-Day Mental Stress Monitoring System using Personal Physiological Data. In Proceedings of the 18th IEEE International Conference on Control, Automation and Systems (ICCAS), Daegwallyeong, Korea, 17–20 October 2018.
48. Donati, M.; Celli, A.; Ruiu, A.; Saponara, S.; Fanucci, L. A Telemedicine Service System Exploiting BT/BLE Wireless Sensors for Remote Management of Chronic Patients. *Technologies* **2019**, *7*, 13. [CrossRef]
49. Gunady, S.; Keoh, S. A Non-GPS based Location Tracking of Public Buses using Bluetooth Proximity Beacons. In Proceedings of the 5th IEEE World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 15–18 April 2019.
50. Kumpdetch, N.; Ayuthaya, S.; Kiattisin, S. Model of Tracking and Identification System in Micro-Location Base. In Proceedings of the 3rd IEEE Technology Innovation Management and Engineering Science International Conference (TIMES-iCON), Bangkok, Thailand, 12–14 December 2018.
51. Rozum, S.; Sebesta, J. SIMO RSS measurement in Bluetooth low power indoor positioning system. In Proceedings of the 28th IEEE International Conference Radioelektronika, Prague, Czech Republic, 19–20 April 2018.
52. Salti, T.; Orlando, M.; Hood, S.; Knelson, G.; Iarocci, M.; Lazzara, Z. A New Set of Bluetooth-Based Fingerprinting Algorithms for Indoor Location Services. In Proceedings of the 9th IEEE Annual Information Technology, Electronics and Mobile Communication, Vancouver, BC, Canada, 1–3 November 2018.
53. Sthapit, P.; Gang, H.; Pyun, J. Bluetooth Based Indoor Positioning Using Machine Learning Algorithms. In Proceedings of the IEEE International Conference on Consumer Electronics—Asia (ICCE-Asia), Jeju, Korea, 24–26 June 2018.
54. GKoufas, Y.; Braghin, S. Anatomy and Deployment of Robust AI-Centric Indoor Positioning System. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops, Kyoto, Japan, 11–15 March 2019.
55. Sevier, S.; Tekeaglu, A. Analyzing the Security of Bluetooth Low Energy. In Proceedings of the IEEE International Conference on Electronics, Information, and Communication (ICEIC), Auckland, New Zealand, 22–25 January 2019.
56. Pallavi, S.; Narayanan, V. An Overview of Practical Attacks on BLE Based IOT Devices and Their Security. In Proceedings of the 5th IEEE International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 15–16 March 2019.
57. Martin, J.; Alpuche, D.; Bodeman, K.; Brown, L.; Fenske, E.; Foppe, L.; Mayberry, T.; Rye, E.; Sipes, B.; Teplov, S. Handoff All Your Privacy—A Review of Apple's Bluetooth Low Energy Continuity Protocol. *Proc. Priv. Enhancing Technol.* **2019**, *2019*, 34–53. [CrossRef]
58. Collotta, M.; Pau, G. A Novel Energy Management Approach for Smart Homes Using Bluetooth Low Energy. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 2988–2996. [CrossRef]
59. Collotta, M.; Pau, G. An Innovative Approach for Forecasting of Energy Requirements to Improve a Smart Home Management System Based on BLE. *IEEE Trans. Green Commun. Netw.* **2017**, *1*, 112–120. [CrossRef]
60. Hong, J.; Levy, A.; Riliskis, L.; Levis, P. Don't Talk Unless I Say So! Securing the Internet of Things with Default-Off Networking. In Proceedings of the IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation, Orlando, FL, USA, 17–20 April 2018.
61. Martinez, C.; Eras, L.; Dominguez, L. The Smart Doorbell: A proof-of-concept Implementation of a Bluetooth Mesh Network. In Proceedings of the Third IEEE Ecuador Technical Chapters Meeting (ETCM), Cuenca, Ecuador, 15–19 October 2018.