computers

MDPI

Article

IoT Security Mechanisms in the Example of BLE

Evgeny Kalinin *, Danila Belyakov, Dmitry Bragin and Anton Konev

Faculty of Security, Tomsk State University of Control Systems and Radioelectronics, 634000 Tomsk, Russia; bds2@csp.tusur.ru (D.B.); bds@csp.tusur.ru (D.B.); kaa1@keva.tusur.ru (A.K.)
* Correspondence: kot60068@vtomske.ru

Abstract: In recent years, a lot of IoT devices, wireless sensors, and smart things contain information that must be transmitted to the server for further processing. Due to the distance between devices, battery power, and the possibility of sudden device failure, the network that connects the devices must be scalable, energy efficient, and flexible. Particular attention must be paid to the protection of the transmitted data. The Bluetooth mesh was chosen as such a network. This network is built on top of Bluetooth Low-Energy devices, which are widespread in the market and whose radio modules are available from several manufacturers. This paper presents an overview of security mechanisms for the Bluetooth mesh network. This network provides encryption at two layers: network and upper transport layers, which increases the level of data security. The network uses sequence numbers for each message to protect against replay attacks. The introduction of devices into the network is provided with an encryption key, and the out-of-band (OOB) mechanism is also supported. At the moment, a comparison has been made between attacks and defense mechanisms that overlap these attacks. The article also suggested ways to improve network resiliency.

Keywords: Bluetooth mesh; BLE; security; IoT

Citation: Kalinin, E.; Belyakov, D.; Bragin, D.; Konev A. IoT Security Mechanisms in the Example of BLE. *Computers* **2021**, *10*, 162. https://doi.org/10.3390/computers10120162

Academic Editor: Sergio Correia

Received: 29 October 2021
Accepted: 26 November 2021
Published: 29 November 2021

1. Introduction

Mesh network–network topology dynamically establishes the maximum possible number of connections between devices for efficient and resilient data transmission [1]. Usually, mesh networks are based on several wireless technologies, such as Wi-Fi, Bluetooth, Thread, and Zigbee [2–5]. However, a mesh network based on Bluetooth Low Energy (BLE) is a popular solution [3,6] due to its high popularity, low cost, and low power consumption [3,7].

The main problem of building mesh networks is to ensure the security of data transmission [8]. Since the network can cover a large area [4], it is necessary to provide efficient protection against unauthorized access by intruders.

The aim of this paper is to compare wireless attacks and defense mechanisms implemented in the Bluetooth mesh standard, which was created by the Bluetooth Special Interest Group (SIG) and implemented on Bluetooth Low-Energy devices starting with 4.0 [9]. The current version of Mesh Profile 1.0.1 can be implemented on BLE devices from 5.0 and later. [10]. Furthermore, one of the tasks is to search for network vulnerabilities and provide recommendations for their elimination.

2. Wireless Network Vulnerabilities

Before considering the existing information security mechanisms, its necessary to introduce a classification of possible network attacks on wireless networks [11–13]:

- Denial of Service. The purpose of this attack is to overload the device with redundant packets, which make the device unusable [14,15];
- Eavesdropping. An attacker eavesdrops on a data exchange in order to extract useful information;

- Man In The Middle, MITM. A malicious device secretly establishes a connection between two devices and making them think they are exchanging data with each other;
- Replay Attack. A previously sent valid message captured by an intruder can be used to exploit the system functionality without an authentication procedure [15].
- Relay Attack. A malicious device establishes communication between two nodes and transmits unmodified data between them [14].

Thus, in order to ensure the information security of wireless devices, it is necessary to take these attacks into account when developing security algorithms. However, these attacks are network attacks and do not include physical interaction attacks on the device, such as attacks to obtain security parameters [16].

3. Bluetooth Mesh Overview

First of all, we need to define the terms used in the Bluetooth mesh specifications [10]. Devices that can transmit or receive messages are called nodes or provisioned devices. Devices that are not a part of any mesh network are called unprovisioned devices. The provisioning process [17] is carried out by a provisioner device, which authenticates unprovisioned devices, assigns an address, and transmits encryption keys [18]. Further configuration of a node is performed by a configuration client, which can be a part of a provisioner device or a part of another node. A configuration client transmits an application key, additional network keys, and configures subscription and publish address for each model.

Each mesh node must have at least one element. An element is an addressable entity that contains models, which defines node functionality. All data exchanges are carried out using messages, which are defined by the opcode, associated parameters, and behavior. Messages are operating with states that represent the position of an element.

There are three types of models [19]: The Server model contains one or several states, divided between one or several elements, as well as messages and behaviors associated with receiving and sending data; the Client Model contains messages necessary for requesting, changing, or using corresponding states of the server; the Control model contains the functionality of both models.

The Bluetooth specification [9] also defines two types of messages: Control Messages for controlling network operations and Access Messages for data distribution. A data exchange between nodes is defined by subscription and publication methods [20,21]. Addresses can be unicast, group, and virtual. Unicast addresses are assigned to each element.

Bluetooth Mesh specification [10] describes the different types of nodes [22]:

- A relay node is a device designed for data transmission within a mesh network. The message forwarding distance is limited by the TTL value;
- A low-power node is a device that spends most of its time in sleep mode. After waking up, a low-power device receives data from a friendly host;
- A friend node is a device that stores data for the Low-Profile node;
- A proxy node is a device that can work with BLE devices via GATT;
- A provisioner node is a device designed to register nodes in the mesh network and to distribute security keys. It also can be a configurator device.

Unlike other protocols (such as Zigbee, Thread, etc.), which are based on the use of routed networks, Bluetooth Mesh uses managed flooding for data transfer within the network [19,23,24]. Managed flooding is a node organizing method within the mesh network, which is a compromise between packets routing and uncontrolled forwarding of received packets. The essence of the method is that incoming packages have a time-to-live (TTL) value, which decreases with each packet transmission until the TTL value becomes zero. Furthermore, all incoming packets are cached to avoid re-forwarding. However, there is an implementation of routed networks based on BLE technologies [25,26].

Bluetooth Mesh is based on Bluetooth Low-Energy technology [10], which works in the 2.4 GHz frequency range. The entire frequency range is divided into 40 channels of

2 MHz each [9,27,28]. Since Bluetooth Mesh usually works without active connections, all communication happening in advertising channels by sending a special non-scanned type of advertising message "ADV_NONCONN_IND" [29]. Interaction between nodes occurs through one of three advertising channels: 37, 38, 39 [14,21].

3.1. Bluetooth Mesh Layers

Bluetooth Mesh has a layered model based on Bluetooth Low Energy. The Bluetooth mesh architecture consists of the following layers [20,21,28,30]:

- The models layer defines model implementation, its behavior, state;
- The foundation models layers defines network configuration and models management;
- The access layers defines application interaction with the upper transport layer to determine data format, data encryption process, and data verification;
- The upper transport layer defines message encryption and verification methods by using the application key generated for each device;
- The lower transport layer performs segmentation of transmitted messages and assembly of incoming messages;
- The network layer defines a message format for transfer data across network elements through a data link layer and message encryption. It also manages messages to be relayed, accepted, or rejected;
- The bearer layer defines packets handling methods such as transmitting data into advertising bearer, which is used in scanning and advertising state. Data transmitting through GATT, which allows communication with regular BLE devices using proxy nodes.

For security reasons, Bluetooth Mesh uses two types of keys for data transmission [28]: AppKey using for data encryption on the upper transport layer, and NetKey for data encryption on the network layer. The same NetKey is used for all nodes within the same network. Key separation allows an intermediate node to verify message integrity and forward it without exposing content, which protects data from unauthorised access. In addition, key separation allows you to secure not only from an eavesdropping attack but also from a relay attack, in which the node that is part of the network can read the messages that are not intended for it [29].

In addition to using AppKey and NetKey, there is a unique key for each device called DevKey. This key is known only to the device itself and the configuration client. DevKey is used for secure communication between the node and the configuration client. Just like the application key, the device key is used in the upper transport layer.

3.2. Authentication and Encryption

Message exchange is secured using the AES-CCM algorithm [10,31]. On the upper transport layer used an application key called AppKey for data encryption and authentication. On the network layer used two security keys: the EncryptionKey for data encryption and authentication and the Privacy key for obfuscation of the message headers. The EncryptionKey and PrivacyKey are divided from NetKey.

Due to the use of the AES-CCM algorithm, all messages have an authentication tag called the message integrity check (MIC).

A unique number that can only be used once (Nonce) is used to encrypt data. Using Nonce protects against replay attacks. Bluetooth Mesh defines four types of Nonce: network nonce (Figure 1), application nonce (Figure 2), device nonce, and proxy nonce.

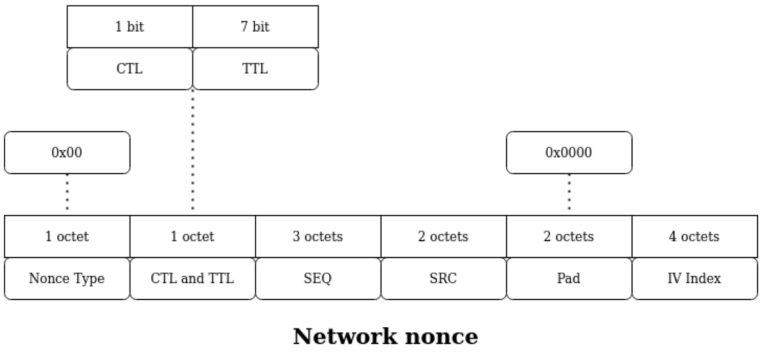

Figure 1. Network Nonce.

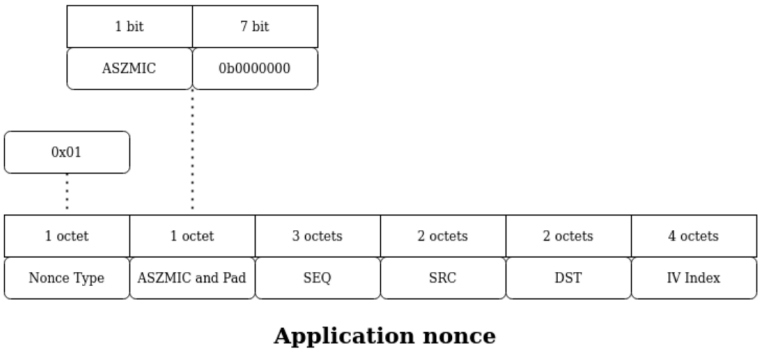

Figure 2. Application Nonce.

A nonce value contains a field type, sequence number (SEQ), source address (SRC), destination address (DST), an initialization vector (IV Index), flag type, TTL value, and ASZMIC flag to indicate the segmented message.

Message headers are obfuscated to hide identifying information, such as source address (SRC), sequence number (SEQ), etc. Header obfuscation provides protection against eavesdropping attacks. The obfuscation process using the AES algorithm with PrivacyKey and an encrypted network message.

Figure 3 presents the generation of a secure network message. The generated network message contains public information about the IVI value, the least significant bit of the IV Index, and network identifier (NID), which is used to determine the encryption key. Network identifier derived from NetKey, EncryptionKey, and PrivacyKey.

Transmitted messages can be eavesdropped on and resent later in an unmodified form. This attack is a Replay attack, in which the same message is transmitted several times, which has a malicious impact on the network. To secure the network from that kind of attack, Bluetooth Mesh defines the sequence number value (SEQ). Nodes increments a sequence value for each message transmission. If a node receives a message with a sequence number lower than previous messages, it will be rejected.

If the sequence number (SEQ) reaches its maximum value, the IV Index update procedure will be started. The IV Index is an equal value for all nodes within a network. The IV Index is updated periodically to avoid reuse, and the update procedure can be launch by any node. The IV Index must be equal to or greater than the value of the next message.

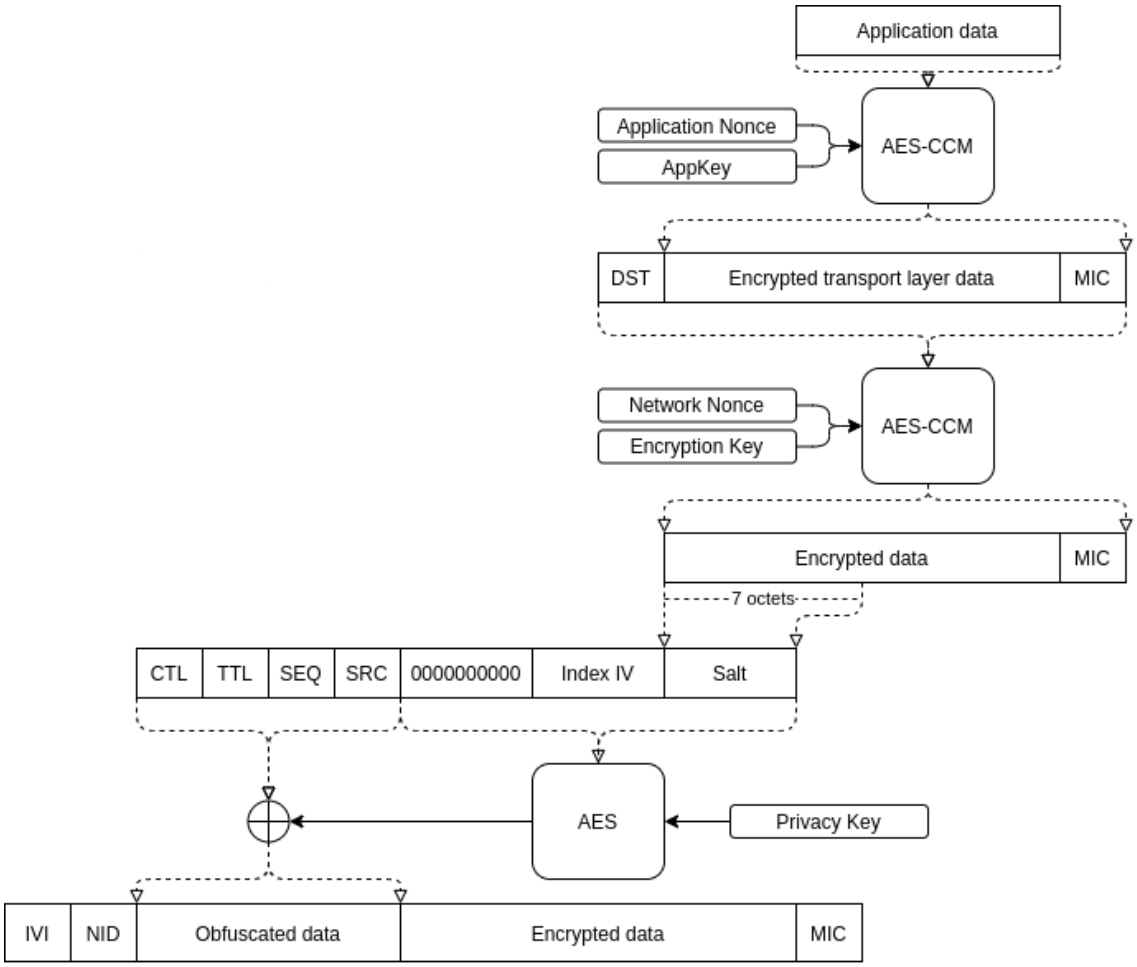

Figure 3. Formation of a secure network message.

3.3. Provisioning Procedure

A mesh network can be created by a provisioner device, which searches for unprovisioned devices and registers them on the network. The provisioner protects the network against using malicious unprovisioned devices.

The provisioning process begins after the unprovisioned device broadcasts advertising messages saying that it is available. When the provision finds the message data, it sends a provision request to the unprovisioned device and information about the supported security algorithms, public key, etc. After the response, the public key is exchanged between the provisioner and the unprovisioned device. Using the Elliptic Curve Diffie–Hellman (ECDH) protocol [32] during the provisioning process, the provisioner sends encrypted security parameters. To prevent an attack man-in-the-middle (MITM), an optional Out-of-Band (OOB) mode [33] can be used (for example, passphrase input, use of NFC technology, etc.). Once a secure connection has been established, the devices exchange provisioning data, such as NetKey, AppKey, Index IV, TTL value, node address, etc.

Discarded nodes must be disposed of, so keys stored within the device cannot be used for attacks on the network. In order to protect against such an attack, the provisioner device adds a disposed-of node to the blacklist, and each node begins the security key update procedure. The provisioner provides new NetKey and AppKey for each node, except devices from the blacklist.

4. Experiment

Table 1 shows a summary comparison of network attacks with Bluetooth Mesh protection methods. As shown, Bluetooth Mesh is vulnerable to denial of service attacks.

Since communications with nodes within a network happen by using only three channels, in an environment with high radio frequency interference, it can cause a significant loss of data. For example, general BLE devices use these channels to notify other devices, and the same channels are used by beacons to continuously broadcast messages [3,5]. Furthermore, the same frequency range is used by other technologies, such as WiFi, ZigBee, etc. [8,23].

Table 1. Comparison of network attacks and protection methods.

Attack Type	Protection Methods
Denial of Service	-
Eavesdropping	Message encryption and message header obfuscation
Man In The Middle	Authentication during node provisioning process by using OOB
Replay Attack	Use of the sequential number (SEQ) and IV Index
Relay Attack	Authentication during node provisioning process by using OOB and using two-level key separation

We have experimentally demonstrated the possibility of denying a service attack on a Bluetooth Mesh network, which continuously transmits sequentially numbered messages to identify lost packets. For this purpose, our experimental setup includes two developer boards nRF52840 [34] from Nordic Semiconductor working within the Bluetooth Mesh network; these devices are also featured in [35] as a low-cost test bed, three SDR ADALM-PLUTO [36] from Analog Devices, which all generate radio frequency interference on the same three channels. The first node sends sequence numbers, and the second node receives them and transmits them to the computer. The ADALM-PLUTO SDR was chosen because it can generate or acquire RF analog signals in a range from 325 MHz to 3800 MHz, and BLE work on 2.4 GHz. The ADALM-PLUTO SDR was handled with GNURadio software in order to control and generate an interference at a specific frequency.

The experiment was carried out in the research laboratory of the Tomsk State University of Control Systems and Radioelectronics. Current research at the time included various wireless devices, operating at 802.11, and BLE standards at the same 2.4 GHz frequency band. This is conducted in order to create as realistic an environment as possible with possible radio interference.

The result is shown in Figure 4, which shows the arrival time of the packet and the channel on which the packet was received. The experiment consisted of five stages, 10 min each: in the first stage, the jammers were turned off; in the second stage, channel 37 was muted; in the third stage, 37 and 38 channels were muted; in the fourth stage, 37, 38, and 39 channels were muted; in the fifth stage 38 and 39 channels were muted. As a result, in the fourth stage, not all packets were drowned; this is explained by the close location of the mesh network nodes, as well as the low power of the jammers. The experiment showed that this network does not have any algorithms for protection against jammer attacks, such as calculating the location of jammers, which are discussed in the article by Dhivyasri et al. [37].

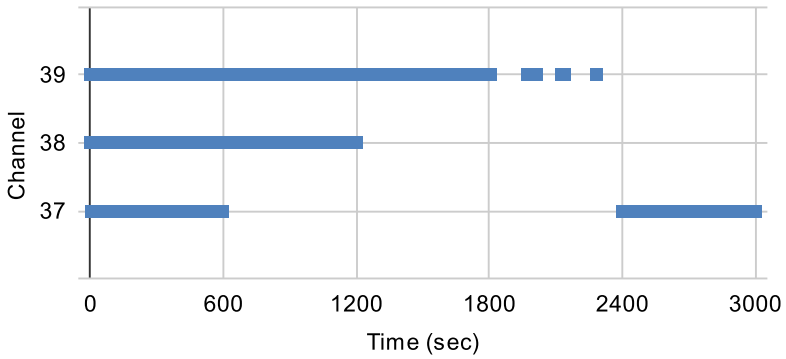

Figure 4. Received packets for each channel.

The following figures show the percentage of packet loss: the percentage of packet loss for each stage is shown in Figure 5, and the dynamics of changes in the percentage of total packet loss are shown in Figure 6.

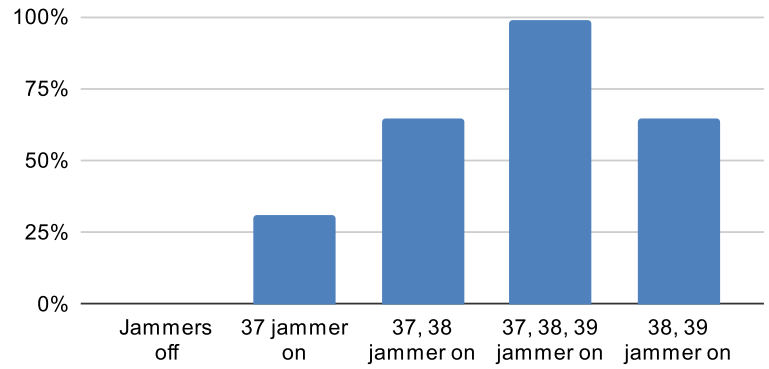

Figure 5. The percentage of packet loss for each stage.

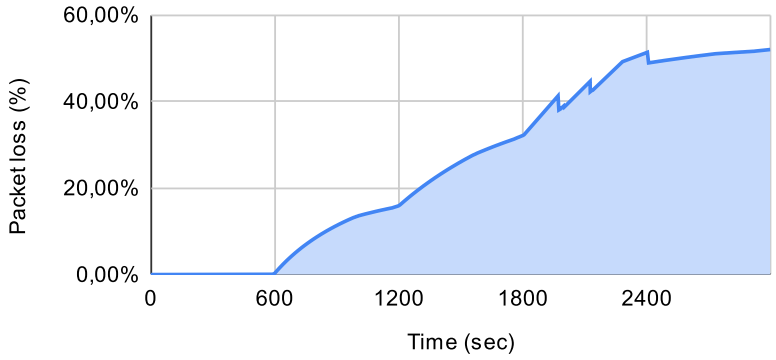

Figure 6. The dynamics of changes in the percentage of total packet loss.

The experiment confirms that with a small amount of equipment, it is possible to disrupt the network. Therefore, packet loss can be critical for systems, such as healthcare systems used to monitor patients' vital signs and their location [38]. The solution is to increase the number of used channels for data transmission, and it is necessary to use all available 40 channels. This can increase the fault tolerance of the network and make it difficult to implement such an attack.

5. Conclusions

The cheapening of microchip production has led to the rapid growth of the Internet of Things and the proliferation of various sensors and sensors. However, due to the large number of devices, it has become more difficult to deploy a wired network to provide communication between different devices. The solution to this problem is to build a wireless network. A wireless network can be deployed without any installation work, and it can exchange data between multiple devices and span large areas. However, a wireless network also has disadvantages. Due to the fact that data in a wireless network are transmitted in a shared radio environment, a collision can occur, an attacker can eavesdrop or send modified data packets, as well as jam data transmission. As a consequence, it is necessary to guarantee the protection of the transmitted data at a high level, as well as to ensure the possibility of data transmission in the presence of interference.

As a result of the work conducted, the main mechanisms of Bluetooth Mesh protection were considered, possible types of attacks on the wireless network were given, and a summary table was compiled reflecting weaknesses in network protection. Based on the table, an attack vector was found. An experiment has been successfully conducted demonstrating

that despite the protection measures provided in the Bluetooth Mesh standard, it is possible to carry out a simple attack that paralyzes the operation of the entire distributed network. To increase the complexity of the attack presented in the experiment, it is necessary for the network to provide data transmission on all available channels and not only on advertising channels, which are quite noisy due to the prevalence of BLE devices and can be easily blocked by three jammers.

In the future, we plan to delve deeper into the security algorithms of the Bluetooth Mesh, as well as implement protection against the attack demonstrated in the experiment.

Author Contributions: Conceptualization, D.B. (Danila Belyakov) and E.K.; methodology, D.B. (Danila Belyakov) and E.K.; software, D.B. (Danila Belyakov) and E.K.; validation, E.K., D.B. (Dmitry Bragin), and A.K.; formal analysis, E.K.; investigation, E.K.; resources, D.B. (Dmitry Bragin) and A.K.; data curation, E.K.; writing—original draft preparation, D.B. (Danila Belyakov) and E.K.; writing—review and editing, D.B. (Danila Belyakov); visualization, D.B. (Danila Belyakov); supervision, D.B. (Dmitry Bragin); project administration, D.B. (Dmitry Bragin); funding acquisition, D.B. (Dmitry Bragin). All authors have read and agreed to the published version of the manuscript.

Funding: The article was prepared as part of the implantation of the «Leading research center (LRC) «Trusted Sensor Systems», financial support provided by the Ministry of Digital Development, Communications and Mass Media of the Russian Federation and Russian Venture Company (RVC JSC) (Agreement №009/20 dated 4 October 2020).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Cardullo, P.; Roio, D. *Mesh Networks*; Wiley Online Library: Hoboken, NJ, USA, 2020; pp. 1–3. [CrossRef]
2. Cilfone, A.; Davoli, L.; Belli, L.; Ferrari, G. Wireless Mesh Networking: An IoT-Oriented Perspective Survey on Relevant Technologies. *Future Internet* **2019**, *11*, 99. [CrossRef]
3. Zanaj, E.; Caso, G.; De Nardis, L.; Mohammadpour, A.; Alay, O.; Di Benedetto, M.G. Energy Efficiency in Short and Wide-Area IoT Technologies—A Survey. *Technologies* **2021**, *9*, 22. [CrossRef]
4. Basu, S.; Baert, M.; Hoebeke, J. QoS Enabled Heterogeneous BLE Mesh Networks. *J. Sens. Actuator Netw.* **2021**, *10*, 24. [CrossRef]
5. Lin, Y.W.; Lin, C.Y. Beyond Beacons—An Interactive Positioning and Tracking System Solely Based on BLE Mesh Network. 2018, pp. 351–362. Available online: https://link.springer.com/chapter/10.1007/978-3-319-65521-5_30 (accessed on 28 June 2021).
6. Dhandapani, K.; Harshavardhan, A.; Kumar, V.; Sunitha, D.; Korra, S. BLE in IoT: Improved link stability and energy conservation using fuzzy approach for smart homes automation. *Mater. Today Proc.* **2021**, in press. [CrossRef]
7. Darroudi, S.M.; Caldera-Sànchez, R.; Gomez, C. Bluetooth Mesh Energy Consumption: A Model. *Sensors* **2019**, *19*, 1238. [CrossRef] [PubMed]
8. Angelov, K.; Sadinov, S.; Kogias, P. Deployment of mesh network in an indoor scenario for application in IoT communications. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1032*, 012004. [CrossRef]
9. SIG, B. Core Specification 5.3. Available online: https://www.bluetooth.com/specifications/specs/core-specification/ (accessed on 1 July 2021).
10. SIG, B. Mesh Profile 1.0.1. Available online: https://www.bluetooth.com/specifications/specs/mesh-profile-1-0-1/ (accessed on 1 July 2021).
11. Sen, J. *Security and Privacy Issues in Wireless Mesh Networks: A Survey*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 189–272. [CrossRef]
12. Ghori, M.; Wan, T.C.; Anbar, M.; Sodhy, G.; Rizwan, A. Review on security in bluetooth low energy mesh network in correlation with wireless mesh network security. In Proceedings of the 2019 IEEE Student Conference on Research and Development (SCOReD), Bandar Seri Iskandar, Malaysia, 15–17 October 2019; pp. 219–224. [CrossRef]
13. Toçilla, A. Overview of Security in Wireless Mesh Networks (WMNs). 2020. Available online: https://www.researchgate.net/publication/341043358_Overview_of_Security_in_Wireless_Mesh_Networks_WMNs (accessed on 28 June 2021).
14. Sevier, S.; Tekeoglu, A. Analyzing the Security of Bluetooth Low Energy. In Proceedings of the 2019 International Conference on Electronics, Information, and Communication (ICEIC), Auckland, New Zealand, 22–25 Jane 2019; pp. 1–5. [CrossRef]
15. Gupta, D.S.; Wani, A.; Kumar, S.; Srivastava, A.; Sharma, D. *Wireless Mesh Network Security, Architecture, and Protocols*; IGI Global: Hershey, PA, USA, 2019; pp. 1–27. [CrossRef]

16. Adomnicai, A.; Fournier, J.; Masson, L. Hardware Security Threats Against Bluetooth Mesh Networks. In Proceedings of the 2018 IEEE Conference on Communications and Network Security (CNS), Beijing, China, 30 May–1 June 2018; pp. 1–9. [CrossRef]
17. Hortelano, D.; Olivares, T.; Ruiz, M.C. Providing interoperability in Bluetooth mesh with an improved provisioning protocol. *Wirel. Netw.* **2021**, *27*, 1–23. [CrossRef]
18. Ghori, M.; Wan, T.C.; Sodhy, G. Bluetooth Low Energy Mesh Networks: Survey of Communication and Security Protocols. *Sensors* **2020**, *20*, 3590. [CrossRef] [PubMed]
19. SIG, B. Mesh Model 1.0.1. Available online: https://www.bluetooth.com/specifications/specs/mesh-model-1-0-1/ (accessed on 1 July 2021).
20. Suthar, F.A.; Patel, R.K.; Prajapati, J.B. Overview of Wireless Mesh Network's in Bluetooth Mesh. 2019. Available online: https://ssrn.com/abstract=3817363 (accessed on 1 July 2021).
21. Baert, M.; Rossey, J.; Shahid, A.; Hoebeke, J. The Bluetooth Mesh Standard: An Overview and Experimental Evaluation. *Sensors* **2018**, *18*, 2409. [CrossRef] [PubMed]
22. Wan, Q.; Liu, J. Smart-Home Architecture Based on Bluetooth mesh Technology. *IOP Conf. Ser. Mater. Sci. Eng.* **2018**, *322*, 072004. [CrossRef]
23. Brandao, A.; Lima, M.; Abbas, C.; García Villalba, L. An Energy Balanced Flooding Algorithm for a BLE Mesh Network. *IEEE Access* **2020**, *8*, 97946–97958. [CrossRef]
24. Darroudi, S.M.; Gomez, C. Bluetooth Low Energy Mesh Networks: A Survey. *Sensors* **2017**, *17*, 1467. [CrossRef] [PubMed]
25. Sirur, S.; Juturu, P.; Gupta, H.P.; Serikar, P.R.; Reddy, Y.K.; Barak, S.; Kim, B. A mesh network for mobile devices using Bluetooth low energy. In Proceedings of the 2015 IEEE SENSORS, Busan, Korea, 1–4 November 2015; pp. 1–4. [CrossRef]
26. Murillo, Y.; Reynders, B.; Chiumento, A.; Malik, S.; Crombez, P.; Pollin, S. Bluetooth now or low energy: Should BLE mesh become a flooding or onnection oriented network? In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017; pp. 1–6. [CrossRef]
27. Zhang, Y.; Weng, J.; Dey, R.; Fu, X. Bluetooth Low Energy (BLE) Security and Privacy. 2019. Available online: http://www.cs.ucf.edu/~rajib/BLE-Encyclopedia2019.pdf (accessed on 28 June 2021).
28. Darroudi, S.M.; Gomez, C.; Crowcroft, J. Bluetooth Low Energy Mesh Networks: A Standards Perspective. *IEEE Commun. Mag.* **2020**, *58*, 95–101. [CrossRef]
29. Hernandez-Solana, A.; Pérez Díaz de Cerio, D.; Garcia-Lozano, M.; Valdovinos, A.; Valenzuela, J.L. Bluetooth Mesh Analysis, Issues and Challenges. *IEEE Access* **2020**, *8*, 53784–53800. [CrossRef]
30. Veiga, A.; Abbas, C. Proposal and Application of Bluetooth Mesh Profile for Smart Cities' Services. *Smart Cities* **2018**, *2*, 1–19. [CrossRef]
31. Padgette, J.; Bahr, J.; Batra, M.; Holtmann, M.; Smithbey, R.; Chen, L.; Scarfone, K. *NIST Special Publication 800-121 Revision 2, Guide to Bluetooth Security*; Technical Report; National Institute of Standards and Technology (NIST): Gaithersburg, MD, USA, 2017. [CrossRef]
32. Diffie, W.; Hellman, M. New directions in cryptography. *IEEE Trans. Inf. Theory* **1976**, *22*, 644–654. [CrossRef]
33. Ren, K. Provisioning a Bluetooth Mesh Network Part 2. 2017. Available online: https://www.bluetooth.com/blog/provisioning-a-bluetooth-mesh-network-part-2/ (accessed on 1 July 2021).
34. Nordic. nRF52840 DK. Available online: https://www.nordicsemi.com/Products/Development-hardware/nRF52840-DK (accessed on 1 July 2021).
35. Murillo, Y.; Reynders, B.; Chiumento, A.; Pollin, S. A Multiprotocol Low-Cost Automated Testbed for BLE Mesh. *IEEE Commun. Mag.* **2019**, *57*, 76–83. [CrossRef]
36. PlutoSDR. ADALM-PLUTO Overview. Available online: https://plutosdr.org/adalm-pluto-overview/ (accessed on 1 July 2021).
37. Dhivyasri, K. Wireless Sensor Network Jammer Attack: A Detailed Review. *Int. J. Res. Appl. Sci. Eng. Technol.* **2020**, *8*, 233–238. [CrossRef]
38. Di Marco, P.; Park, P.; Pratesi, M.; Santucci, F. A Bluetooth-Based Architecture for Contact Tracing in Healthcare Facilities. *J. Sens. Actuator Networks* **2020**, *10*, 2. [CrossRef]