Technische Universität Ilmenau

Fakultät für Elektrotechnik und Informationstechnik

Media Project

Performance Evaluation of IPv6 Communication over Bluetooth Low Energy

---

submitted by:	Umer Mahmood, Qasim Ali
submitted on:	11. 11. 2020
Born on:	28. 12. 1991 in Kotli , 06. 07. 1990 in Lahore
Major:	Master Media Technology
Studienrichtung:	Fakultät für Elektrotechnik und Informationstechnik
Anfertigung im Fachgebiet:	Kommunikationsnetze
Verantwortlicher Professor:	Prof. Dr. rer. nat. Jochen Seitz
Wissenschaftlicher Betreuer:	Dr. Silvia Krug

Abstract

Bluetooth Low Energy (BLE) is part of the current Bluetooth standard that was specially developed for setting up networks with low energy consumption and high reliability. In addition to the traditional variant, an option for IP-based networks was also created. This enables further fields of application for BLE in the area of wireless sensor networks and the Internet of Things. In order to be able to make a decision on the use of IP-based BLE networks, precise knowledge of their performance characteristics is necessary. The main goal of this project is to evaluate, the performance of a BLE-based IPv6 network and assess on the basis of practical experiments using an existing implementation by Nordic Semiconductors. In particular, connection stability, data throughput and long-term behavior are considered. For these assessments, the underlying BLE connection parameters are configured in various combinations using one of the leading BLE enabled System on Chip (Soc) NRF52. A small network of NRF52 nodes and a Raspberry pi as a router is setup. One of the nodes is configured as a UDP server while other nodes act as clients. Using this setup, data is transferred between the server and clients, while measuring throughput, latency and connection stability. The measurements in these experiments show a maximum of 166 kbps of effective throughput and 14 ms of latency can be achieved with Bluetooth versions 4.2 and above. Connection stability remained good, i.e, no disconnection events were noted, with any of the used BLE connection parameters. These findings are significant for future IPv6 over BLE projects. Based on these measurements, the suitablity of IPv6 over BLE can be considered for specific applications in future. Moreover, based on the theoretical comparison of BLE, classic Bluetooth and Bluetooth Mesh, a decision on the choice of one of these technologies in future projects can be made.

Contents

1 Introduction

Internet Protocol version 6 (IPv6) and Bluetooth Low Energy (BLE), both have been around for quite some time now. IPv6 is the most recent of the internet protocols, which provides a system for identification and location of devices on the internet moreover providing mechanism for routing data traffic. Where as bluetooth is the technology which provides connectivity to devices using wireless medium over short distances using the 2.4 Ghz ISM band. BLE is one of three kinds of bluetooth technologies, the others being classic bluetooth and Bluetooth Mesh. BLE is tailored for low power consumption devices which require only small amount of data transfer capabilities, and are expected to run on very low power sources, like, coin cells.

For the devices that have a Wifi module on board, be it sensors or consumer electronics, they can easily connect to the internet to share data. Where as, for applications where low power is desired, Wifi modules cannot be used since they consume more power as compared to BLE devices. BLE based devices connect in a slightly different manner. BLE devices, such as environment sensors or health monitoring devices, require a hub in some cases a smart phone or some other central device, to send data to the internet. IPv6 over BLE, solves this problem in a very elegant way. Using this technology, a network of sensors or devices, can connect to a BLE equipped router, and get connected to the internet and have their unique IPv6 address based identity. Utilizing best of both the technologies, i.e huge address space from IPv6 and low power nature of BLE, IPv6 lays perfect ground for Internet of Things. The sensor network can not only communicate data to a server over internet, but also communicate to each other using their IPv6 addresses. This study focuses on performance evaluation of such a network based on an existing implementation done by Nordic Semiconductors.

1.1 Motivation

According to statistics, among google users worldwide, adoption rate for IPv6 is around 30 percent [goo]. On the other hand, 4.0 Billion Bluetooth devices were shipped in only 2019 [SIG19]. These stats clearly show the wide usage and availablity of both of these technologies. While theoretical analysis of the performance of IPv6 over BLE already exists, there is still need to practically evalute the performance of such a network. Moreover, theoractical and practical evaluation of BLE communication already exist and shows a great potential in terms of effective throughput [Afa20]. Moreover, with ever increasing demand of Internet of Things, absence of practical evaluation of this particular technology restricts its usage in upcoming applications. This motivates the topic of this study to implement a flexible test bed where various parameters this technology can be measured. The goal of this study is to practically measure throughput, latency and connection stability accross different Bluetooth versions and under multiple underlaying BLE connection parameters. A test setup was developed based on a small network of a BLE enabled router and multiple BLE nodes connected to it, exchanging data using IPv6 over BLE. Using this practical setup, following important questions were answered: What is the effective throughput and latency achievable under various BLE connection parameters? How stable is the connection? How various bluetooth versions and bluetooth technologies differ from each other?

1.1.1 Outline

Chapter 1 covers introduction of the the topic and explains motivation behind it. Chapter 2 explains in detail the fundamental components and architecture of IPv6 over BLE technology. Moreover, working of different layers is discussed. Chapter 3 focuses on setup of the test bed developed for various evalutation experiments. Theoratical and measured results of only a BLE connection, as well as an IPv6 connection running over it are presented in it. It also presents affects of changing BLE connection parameters on overall throughput and latency of data transfer using IPv6. Chapter 4 presents a comparison of features of various Bluetooth versions. In chapter 5, a comparison of three bluetooth technologies i.e, Classic Bluetooth, Bluetooth Low Energy and Bluetooth Mesh is presented. Furthermore, their areas of application are also identified.

1.1.2 Definitions

The low power version of Bluetooth included in blueooth specification since version 4.0, which is designed for applications requiring low power consumption, lower data transfer rates and low duty cycles, is called Bluetooth Low Energy [spe13]. The most recent version of Internet Protocol is refered to as IPv6. It mainly offers larger address space and Header Simplification among other new features [HD98]. The transmission of IPv6 packets using Bluetooth Low Energy Link with the help of 6LoWPAN intermediate layers is called IPv6 over BLE [NSI+15]. 6LoWPAN link is wireless link within 1 hop reach ability. 6LoWPAN Node (6LN) is any device participating in a LoWPAN connection and a 6LoWPAN Boarder Router provides routing between two LoWPAN networks or between a LoWPAN and an IP network [BSCN12].

1.1.3 Problem Statement

This study is being carried out to evaluate the performance of IPv6 over BLE network. Studies regarding theoretical analysis of this kind of network have already been carried out but an evaluation using practical implementation is still lacking. Moreover, there exists a great amount of work already to analyses the performance of the underlying Bluetooth Low Energy technology both in theory and practical implementation. In this study we implement a test bed to answer following main questions:

- How does an IPv6 over BLE perform in terms of throughput, latency and connection stability?
- How do different specification versions of BLE differ from each other?
- How do Bluetooth Classic, Bluetooth Low Energy and Bluetooth Mesh differ from each other?

2 IPv6 Over BLE

For the devices which are not constrained in terms of power consumption, for example devices that do not require to run on batteries or are not supposed to have long battery life, networks consisting of WiFi combined with IPv4, have been used for decades now. But in case of low power application domains, for example sensor networks, it is important to have a extremely low power wireless technology and in order for these networks to scale, it is also critical to have a robust mechanism for addressing. IPv6 clearly solves the problem of limited address capacity of IPv4, and combined with BLE, low power scale-able sensor networks are now possible. Further more, IPv6is also equipped with stateless address auto configuration which in particular suits very well to the devices constrained in terms of processing power. Implementation of IPv6 is possible, with the help of 6LoWPAN, which was initially developed to provide specifications for transmission of IPv6 packets over IEEE 802.15.4 networks. Considering the similarity of BLE with IEEE 802.15.4, techniques used in 6LoWPAN are also used for IPv6 over BLE. Since BLE, to maintain its low power properties, does not allow large protocol headers. IPv6 having a header size of 40 bytes, it does not go well with small protocol header design philosopy of BLE. That problem is solved by 6LoWPAN by using header compression techniques. Moreover, techniques of stateless autocnfiguration, link-local IPv6 addresses and neighbor discovery are also used from 6LoWPAN [NSI+15].

2.0.1 Address Space

IPv6provides a huge address space by increasing address size from 32 bits to 128 bits. With long address field, it now supports a great number of uniquely addressable devices as compared to it predecessor with an increased level of hierarchy as well. This increase in address space is ideal for ever growing network of Internet of Things and Industry 4.0 applications. Moreover IPv6 also improves scalability of multicast addresses and introduces anycast address which can be used to address any one group of devices. [HD98]

2.0.2 Stateless address autoconfiguration

Stateless address autoconfiguration referes to the ability of a host to assign itself a unique address in the network with the help of information that is available locally and the information it has received from a router.

2.1 Architecture of IPv6 over BLE

IPv6 comprises of layers from both the technologies. Figure 2.1 illustrates layers and protocols that form IPv6 over BLE stack from Nordic Semicondutors. Since the test bed that would be describe in upcoming chapters, is based on this particular stack implementation, it is important to briefly go through it. The figure 2.1 represents the layers with reference to the OSI model [ipv].

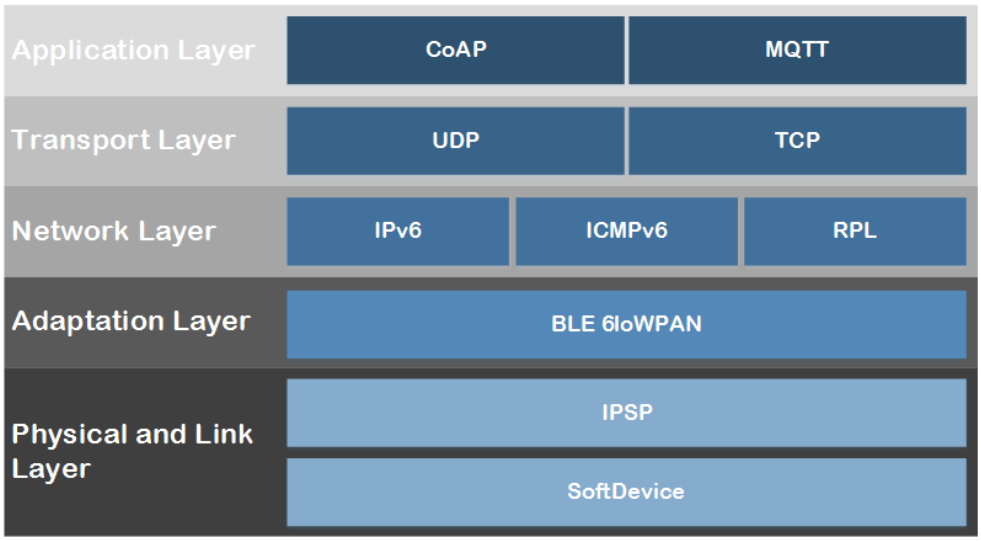

Figure 2.1: IPv6 over BLE `Layers`

2.1.1 Application Layer

In application layer we can see **Constrained Application Protocol (COAP)** and **Message Queuing Telemety Transport (MQTT)**. These are among some of the application protocols that are widely used in constraint envoirnments. But the stack can be used by any other protocols that are capable of running over IPv6[ipv].

2.1.2 Transport Layer

In transport layer as we are familiar, protocols are responsible for process to process communication. We can see widely used protocols **User Datagram Protocol (UDP)** and **Transmission Control Protocol (TCP)** in figure 2.1 . Referring to the 2.1.2, it can be seen that, UDP adds 8 bytes of header data to the user data.

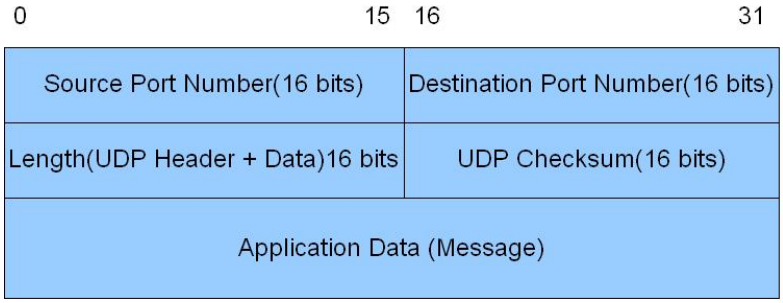

Figure 2.2: IPv6 over BLE `Layers` [Com06]

2.1.3 Network Layer

Network layer is responsible for communicating data between multiple links. It handles neighbor discovery, reachablitlity of devices and routing. From the Figure 2.1 we can see IPv6 , ICMPv6 and RPL form the network layer. In our case we are most concerned with IPv6.From the structure of IPv6 header in Figure 2.1.3 we can see IPv6 adds 40 bytes of header data. Of which major part consists of 16 bytes of source and 16 bytes of destination IPv6 address.

2.1.4 Adaption Layer - BLE 6LowPAN

6LoWPAN plays a vital role in IPv6 over BLE stack. It acts as a adaption layer between Network Layer and Bluetooth specific layers. It performs two important tasks. One, it provides compression of IPv6 and UDP headers, which for IPv6 we have seen in sections 2.1.2 and 2.1.1 is of 40 bytes and for UDP is 8 bytes. These header lengths are not ideal for a BLE link. 6LoWPAN performs header compression according to techniques define in RFC6282. In the best case scenario it reduces IPv6 over BLE down to only 2 bytes using LOWPAN IHPC encoding and UDP header to only 2 bytes

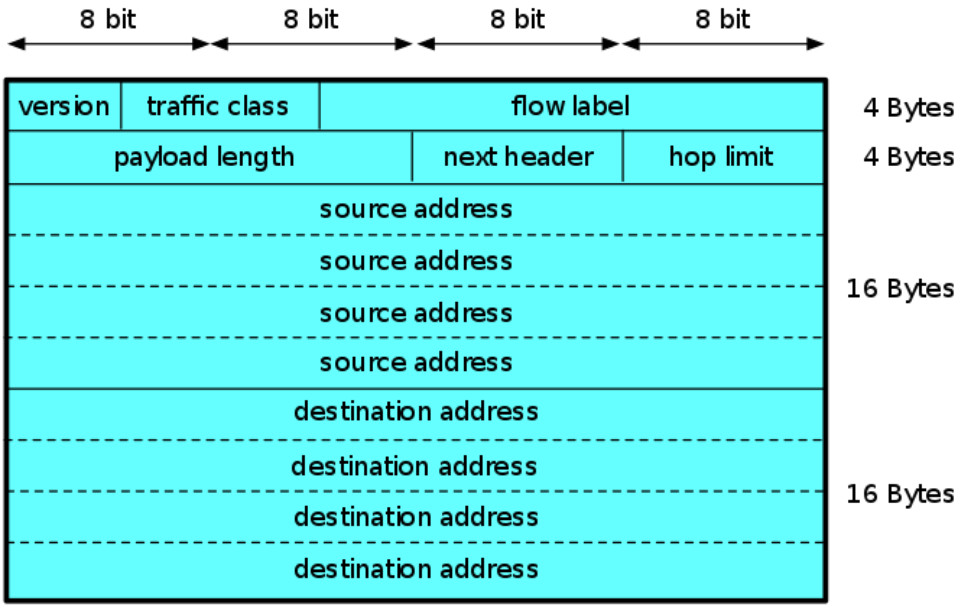

Figure 2.3: IPv6 over BLE `Header Format` [Com11]

using LOWPAN NHC, further it adds 1 byte of context identifier. [CCG18]. So as minimum, 6LoWPAN contains only 5 bytes of header after compression. Complete format of 6LoWPAN layer header can be seen in table 2.1. Secondly, 6LoWPAN adaption layer also provides optimizations for various procedures. For instance, optimization of Neighbour Discovery procedure on Bluetooth Low Energy.

6LoWPAN also provides stateless address autoconfiguration. That is, it provides procedure for generation of network interface link-locak addresses. This consists of two parts, a prefix shared by the router and an interface identifier generated by the hosts [NJT07]. In case of IPv6 over BLE this happens during interface initialization phase. The router shares prefix with the hosts using specific messages, called Router Advertisements (RA). In case of link local addresses, first, using 48 bit Bluetooth address of the nodes is transformed into a 64 bit EUI-64 addrees, which becomes IPv6 MAC address. It is achieved by adding two additional octets. Then, this 64 bit identifier is transformed into an IID by setting universal/local bit to zero. Finally, a pre specified prefix **0xFE80::/10** is added to the IID. Whereas, in case of global address, prefix shared by the router to IPv6 BLE nodes, is a global prefix. In stateless address autoconfiguration, avoiding duplication of addresses is handled by the border router, which also contributes to low power consumption and low complexity for the 6LoWPAN node.

0							8							15
0	1	1	TF	NH	HLIM	CID	SAC	SAM		M	DAC	DAM		
SCI			DCI			1	1	1	1	0	C	P		
SRC PORT			DST PORT											

Table 2.1: Compressed UDP/IP Header Format defined by RFC6282

2.1.5 IPSP

IPSP is responsible for handling how to setup a BLE connection over which 6LoWPAN can function. IPSP defines how 6LN and 6LBR behave in a connection. Moreover, it provides the nodes the ability to be discover able by a 6LBR with the help of GATT protocol. It also defines how fragmentation is handled in the link layer and recommends an MTU of 1280 bytes [CCG18].

2.1.6 Physical and Link Layer

Physical and Link Layer part of IPv6 over BLE stack is composed of two sub category of layers. IPSP and Bluetooth Low Energy Link Layers. Header format of various BLE layers, as per version 5.0 can be seen in figure 2.4.

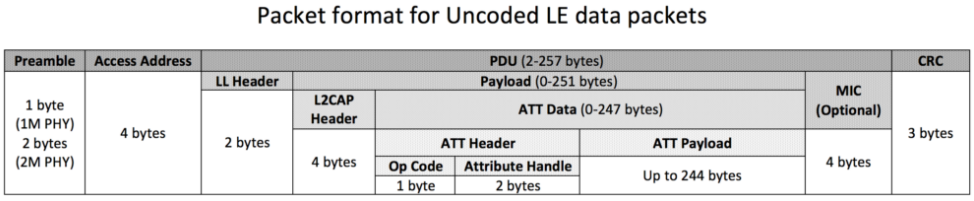

Figure 2.4: BLE Complete `Header Format Based on v5.0` [Afa20]

2.1.6.1 L2CAP Layer

Logical Link Control and Adaption Protocol (L2CAP) layer provides multiplexing so that multiple applications can share the logical link. It does so, by providing channel end points on each device with a Channel Identifier (CID) and is fully responsible for managing those channels. L2CAP is also the layer which provides Fragmentation and

Reassembly to allow transmission of larger data packets [spe13]. As can it can be seen in table 2.2, L2CAP adds 4 bytes of header data. L2CAP layer can have two different kind of frame namely

- Data Frames : used for data transfer.
- Signaling Frames: used for infromation sharing while connection establishment or for updateing connection parameters.

2 bytes	2 bytes	
Length	Channel ID	Information Payload 0 -65535 bytes

Table 2.2: L2CAP LE Data Frame

2.1.6.2 Link Layer

Link layer provides and abstraction between the physical layer and upper layers. Its provides management of radio and takes care of operations such as CRC, random number generation and encryption. Link layer controls various states a device can be in. A BLE device can be in following states during operation:

- Advertising. broadcasting information without being connected to any peer device
- Scanning: listening to advertised information from other devices
- Connected: In a state of established link with a peer device where both the devices continuously share data using agreed upon connection parameters.

The format of link layer header depends on the kind of PDU. It can be an advertising PDU or a Data PDU. The data PDU's are exchanged when devices are in connected state.

2.1.6.3 Physical Layer

Physical layer contains the hardware responsible for wireless communication. It is responsible for converting digital bits into radio waves using the 2.4 Ghz ISM band. The radio divides this spectrum into 40 2Mhz wide channels. Three Channels 37, 38 and 39 are used for advertising while the rest are use for data transfer.

3 Test Bed Development and Performance Evaluation

3.1 Test Setup

The test setup for evaluation of IPv6 over BLE includes major hardware devices

1. Nordic nrf52840 SoC as 6LoWPAN nodes.
2. Raspberry Pi 3b (Bluetooth v4.0) and Raspberry Pi 4 (Bluetooth v5.0) as 6LoWPAN border routers used interchangeably to test on different Bluetooth versions.

On the software side we have

1. Modified nRF5 Software Development Kit from Nordic Semiconductors providing IPv6 stack implementation.
2. Linux based Raspbian Operating System for Raspberry Pi.

Nordic devices use software development kit from Nordic Semiconductors, which includes soft-devices in binary form as bluetooth stack. Whereas, on Raspberrypi Bluez provides the bluetooth stack. The evalutation tests are performed on network structure shown in figure 3.1.

One of the nrf nodes is running a UDP server which is configured to respond with the same data as received. For some of the tests raspberrypi acts as a UDP clients and communicates to the nrf node based UDP server, whereas in some tests, another nrf node acts as a client. Each nrf node has an IPv6 address based on its bluetooth address according to this addressing process(add link here).

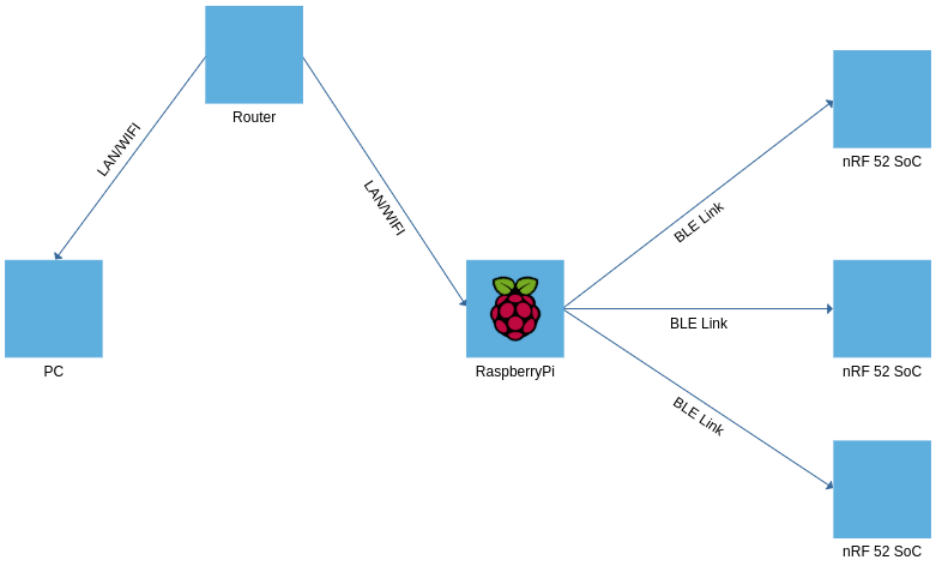

Figure 3.1: Network `Diagram`

3.2 BLE Connection Parameters

A BLE connection has many parameters that are exchanged when a connection is being established. When connection is being established connecting devices exchange their feature list and depending on the list, connection parameters are also exchanged. These connection parameters greatly affect throuput and latency of a BLE connection and hence directly affect throughput and latency of IPv6 data transfer.

1. Phy used (1M or 2M).
2. ATT MTU
3. Connection Interval
4. Gap Event Length
5. Slave Latency
6. Supervison Time Out

These configurations can be changed in order to achieve a balance between throughput and power consumption.

3.2.1 ATT MTU

ATT MTU referes to the maximum data unit that can be packed into a radio packet. By default this unit is set to 23 bytes in software development kit under use. Longer MTU lenghts offer better throughput.

3.2.2 Data Length and Data Length Extension

Data length extension is a feature that was introduced in Bluetooth version 4.2, it allows larger link layer mtu, i.e an increase in length of the data that can be transfered inside one radio packet. Upto v4.1 data length was limited to just 27 bytes, whereas, with DLE, it goes upto 251 bytes. Data length extension has significant effect on througput of BLE connection and hence on IPv6. The fragmentation and reassembly in L2CAP layer, is also directly affected by this. Larger the link layer mtu, lesser the requirement to fragment data packets from upper layers.

3.2.3 Connection Interval

The connection event is the time within a timing-event reserved for sending or receiving packets [ble]. To understand connection interval it is important to first understand **connection events**. A repeated and continuous exchange of packets between master and slave is referred to as a connection event. Such an event can have a exchange of one packet or multiple packets from each side. The start of these events is spaced out by a specific duration of time called **Connection interval**. This interval basically determines when the connected devices listen on the radio. A short interval means, both the devices listen frequently on the radio where as a long interval means they listen less often and keep the radio off for longer time. This directly affects throughput and power consumption. Connection interval can be between **7.5** and **4000** milliseconds and can be varied in units of 1.25 ms i.e the range is (6 to 3200 units). Since within connection intervals we have connection events, so when increasing this value, more packets may be sent in one interval, but if a packet is lost, the wait until the re-transmission is longer. Increasing this value can increase throughput, provided that the GAP event length increases by the same amount or connection event length extension is enabled.

3.2.4 PHY

Over the data rate also depends on which phy is being used. Three phy option are available and are offered by the stack implementation by SDK being used in this study.

- PHY 1 Mbps
- PHY 2 Mbps which was included since Bluetooth version 5.0
- PHY codded which is used for long range feature but with low data rates.

But in this study, the 6LoWPAN border router, the raspberry pi being used, does not have PHY 2 Mbps despite having Bluetooth version 5, since its an option feature. So, tests using 2 Mbps phy are not performed.

3.2.5 Connection Event Length Extension and GAP Event Length

Connection event length extension determines if it is allowed to increase connection event length within a connection interval. Whereas, GAP event length refers to the duration of exchange of packets between two connected devices during a connection interval. A longer connection event allows sending more packets within one connection event before the radio turns of, hence it allows increase in throughput as well. In order to get maximum throughput on a BLE link an it is important on enable conncetion event length extension and increase connection interval [ble].

3.2.6 Slave Latecy and Supervision Timeout

Slave latency defines the maximum number of connection events a slave in a BLE connection can skip, i.e to not respond. Because in normal operation slave is required to respond to data packets sent from master even if it doesn't have any data to send. Whereas, supervision timeout is the maximum time, between two received data packets, after which connection is considered lost. It is calculated by equation 3.2 or equation 3.2 if connection is created but is yet to be established [Afa18].

$$supervisionTimeOut < (1 + slaveLatency) \cdot connectionInterval \cdot 2 \tag{3.1}$$

$$supervisionTimeOut = connectionInterval \cdot 6 \tag{3.2}$$

Test Number	DLE	ATT_MTU (bytes)	Connection Interval (ms)	Phy (Mbps)	Throughput Result
1	False	23	7.5	1	191.48
2	False	23	7.5	2	274.85
3	True	158	7.5	1	285.72
4	True	247	7.5	2	1028.27
5	True	247	50	2	1265.03
6	True	247	400	2	1252.00

Table 3.1: BLE Link throughput at various

3.3 Throughput Test BLE

Before we test performance of an IPv6 network, it is important to test the underlaying BLE link, to get an understanding how does the link behave with various combinations of parameters explained in section 3.1. This would be helpful to estimate expected performance characteristics of a IPv6 network. The structure in figure 3.2 is used to modify these BLE link parameters for various tests that follow. The results of several test for just the BLE link throughput can be seen in the table 3.1 .

```
typedef struct
{
uint16_t        att_mtu;
uint16_t        conn_interval;
ble_gap_phys_t  phys;
uint8_t         data_len;
bool            conn_evt_len_ext_enabled;
uint16_t        slave_latency;
uint16_t        slave_timeout;
uint16_t        gap_evt_len;
} test_params_t;
```

Figure 3.2: BLE connection test parameters `structure`

3.3.1 Discussion on Results

Simple BLE link throughput measurements are promising , in that they follow the expected trend of increase in throughput when making changes in max mtu size, connection interval and datalength extension. But the values are far from the theoretical calculations and also a slightly different from measurements results published in [Afa20].

3.4 IPv6 Over BLE Throughput Tests

In the Throughput test udp protocol is used for transport. Test setup includes a PC and raspberry pi connected to a standard router. Further to the raspberry pi nrf52 node is connected, running as a udp server. The server is configured to respond to udp messages such that it returns the same messages back. For throughput testing, a PC with a python script acts as udp client. Throughput is tested for one Kilobytes of data, with various BLE link configurations. For example, different values of att-mtu, connection interval and data-length-extension on/off.

3.4.1 Test Bed Design Problems

During the development of test bed for calculation, two major problems were faced. First, communication of Raspberry Pi version 3b with Bluetooth v4.1 was successful with nrf52 node, without any modifications to the raspian OS or to the NRF5 SDK. Whereas, when test setup for Data Length Extension feature was required, a raspberry pi with version above 3b with a Bluetooth version above 4.1, was required. For these experiments, a raspberry pi 4 with Bluetooth 5.0 was used. But, the connection between raspberry pi and nrf52 disconnected after just 1 minute. This was resolved after over-the.air inspection of messages exchanged during connection setup, with the help of a ble sniffer in between. The problem turned out to be, nrf52 nodes not replying to the Link Layer Length Request messages from raspberry pi as seen in image 3.3. This was resolved by modifying the ble events handler in Nordic SDK, and triggering a proper response to the length request message from the master. Secondly, when testing various data length configurations, it wasn't possible, to increase tx data length beyond 204 bytes, and rx data length beyond 54 bytes. The problem turned out to be restrictions based on tx and rx buffer lengths in IPSP service configurations in NRF5 SDK. A full data length of 251 bytes was acheived, after modifying those buffers accordingly.

3.4.2 Effect of Connection Interval Measurements on Throughput

This experiment is based on Bluetooth version 4.1 . The throughput measurements are performed with various values of connection interval, without event length extension. Four different values of connection interval were used whereas, slave latency was kept constant at 6. The value of supervision timeout was calculated using equation 3.2 for each case. The figure 3.4 shows the results. It is clear from the graph that effective

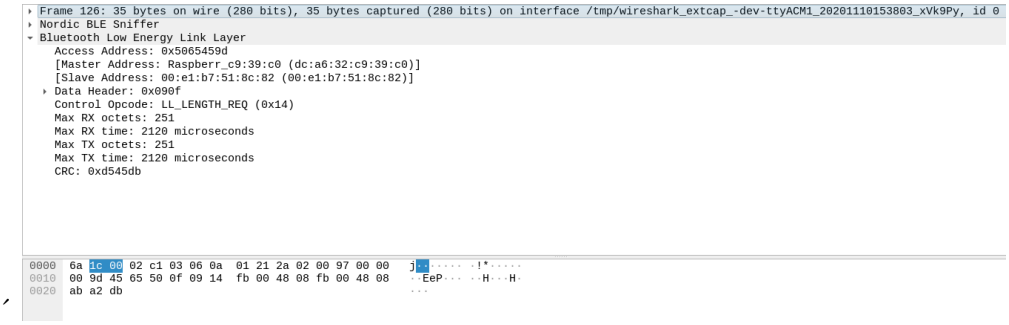

Figure 3.3: Connection Interval vs Throughput Measurements

throughput decreases with increase in connection interval. Furthermore, experiments

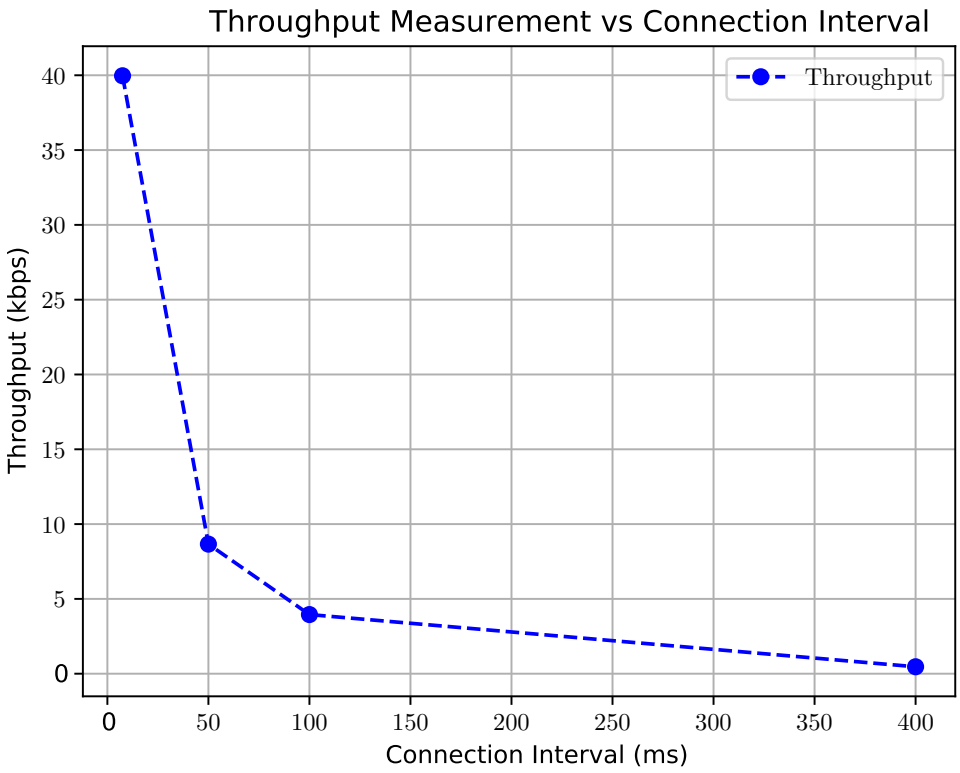

Figure 3.4: Connection Interval vs Throughput Measurements

with Event Length Extension were also performed, on Bluetooth version 5.0 . In these experiments, connetion interval values were varied and data event length was set equal to connection interval for each case. Measured throughput results are show in figure 3.5.

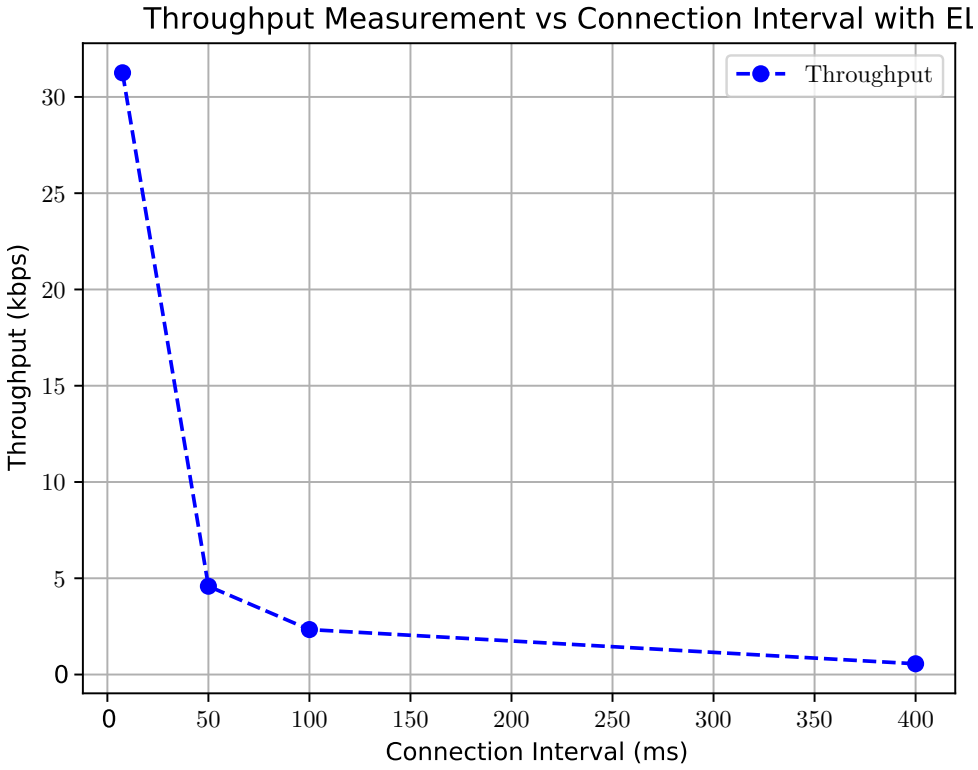

Figure 3.5: Connection Interval vs Throughput Measurements with ELE

3.4.3 Effect of Data Length on Throughput

In these experiments throughput is being measured by changing Data length. These experiments utilize data length extension feature of Bluetooth 4.2 above. Throughput was measured with various data length values while having a constant value of connection interval set at 7.5 ms. Graph 3.6 represents the results. It can be seen throughput increases with increase in Data length upto 162 bytes, but falls down after that.

3.4.4 Effect of Slave Latency on Throughput with Maximum Data Length

In these experiments the effect of different values of slave latency is under observation. In these experiments, data length extension is enabled and data length is set to its maximum value of 251 bytes. Which means, 247 user data bytes can fit into one link layer packet. It should be observed, supervison timeout also changes as per equation 3.2 whereas, connection interval is also kept constant at 7.5 ms. The results of experiments can be seen in figure 3.7. It can be seen, a maximum value of 164 kbps is acheived

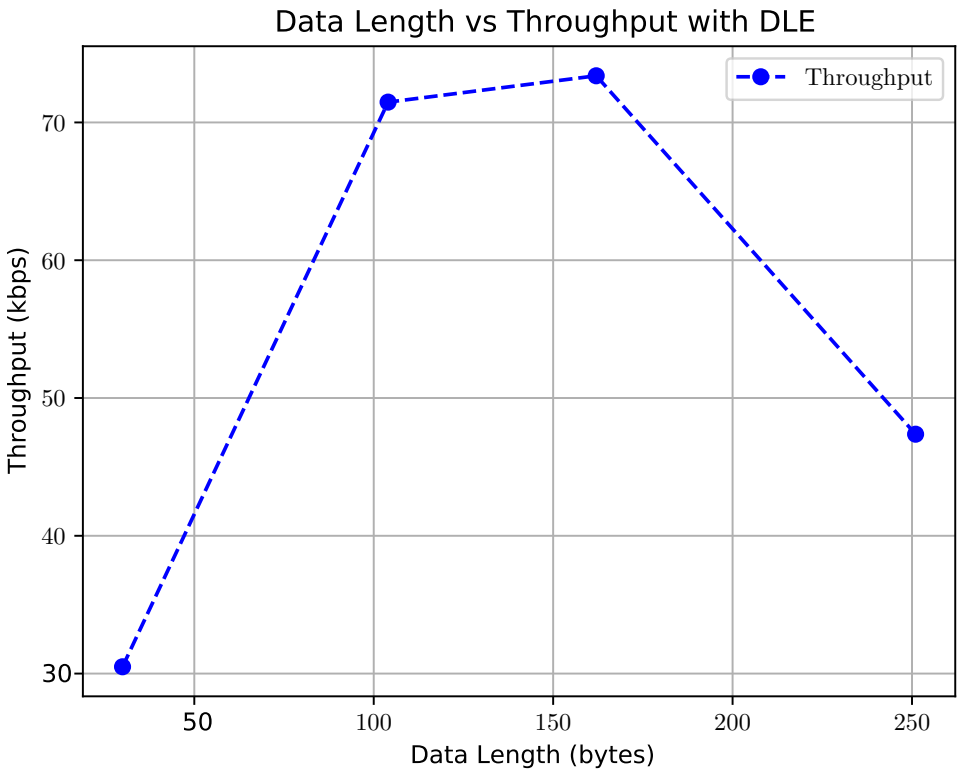

Figure 3.6: Throughput Measurements with Different Datalengths

when slave latency is minimum i.e 0. Furthermore, decrease in throughput is observed with increasing slave latency.

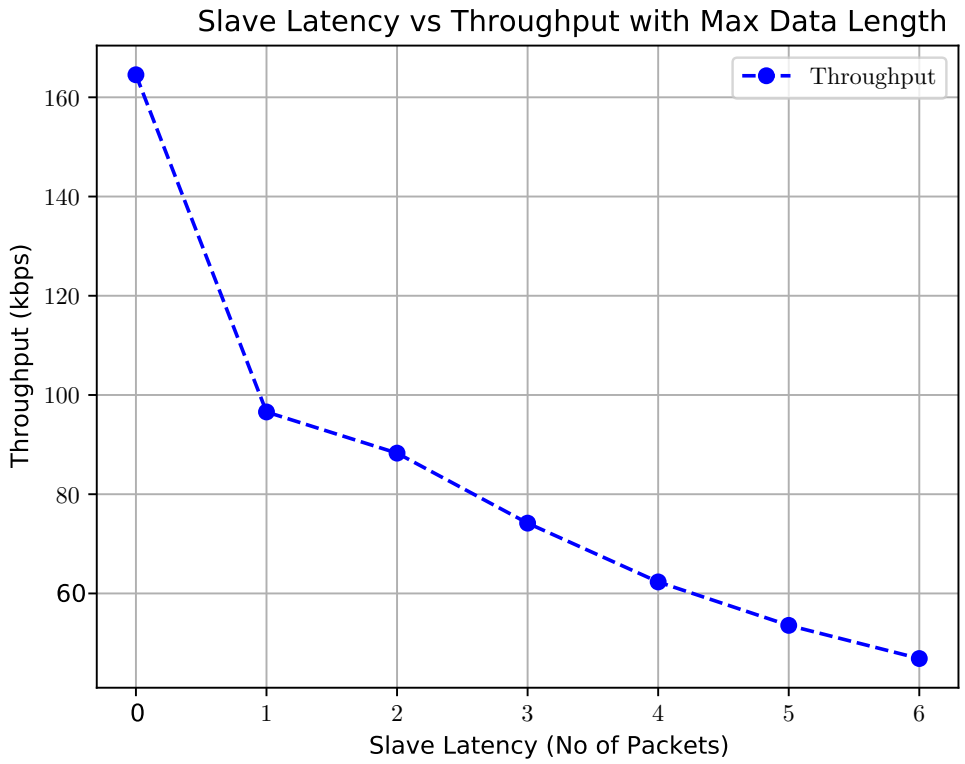

Figure 3.7: Throughput Measurements with Different Datalengths

3.5 Latency Test

For latency measurement, same network configuration exists. To calculate round trip latency, a data packet from border router i.e Raspberry Pi is sent to the UDP server. This data packet contains departure timestamp. The UDP server, returns the same data back. On arrival, The timestamp is read back from the UDP packet, and the difference to the current time is calculated. This gives us the round trip latency. The measurement is performed for 1000 packets. To get a good understanding the test is performed using two configurations:

- Default Configuration: Configuration with all the connection parameters set to their default values as in table 3.2.
- Maximum Configuration: Configuration withh all the BLE connection parameters set their throughput optimum values as in table 3.3.

1	Parameter	Value
2	att_mtu	27 bytes
3	Connection Interval	30 ms
4	Phy	1 Mbps
5	Data Length	27 bytes
6	Connection Event Extension	false
7	Slave Latency	6
8	Supervision Timeout	430 ms
9	Gap Event Length	7 ms

Table 3.2: Default BLE Connection Configuration

1	Parameter	Value
2	att_mtu	251 bytes
3	Connection Interval	7.5 ms
4	Phy	1 Mbps
5	Data Length	251 bytes
6	Connection Event Extension	True
7	Slave Latency	0
8	Supervision Timeout	400 ms
9	Gap Event Length	400 ms

Table 3.3: Maximimum BLE Connection Configuration

3.5.1 Measurements

As can be seen from results in figure 3.8, for default configuraition the measured latency falls around 97 ms with some packets showing a deviation from average value and

reaching maximum of 190ms. Whereas, for the maximum configuration, the observed latency lies on average at 14 ms, and some packets depict a maximum latency value of 23 ms as well.

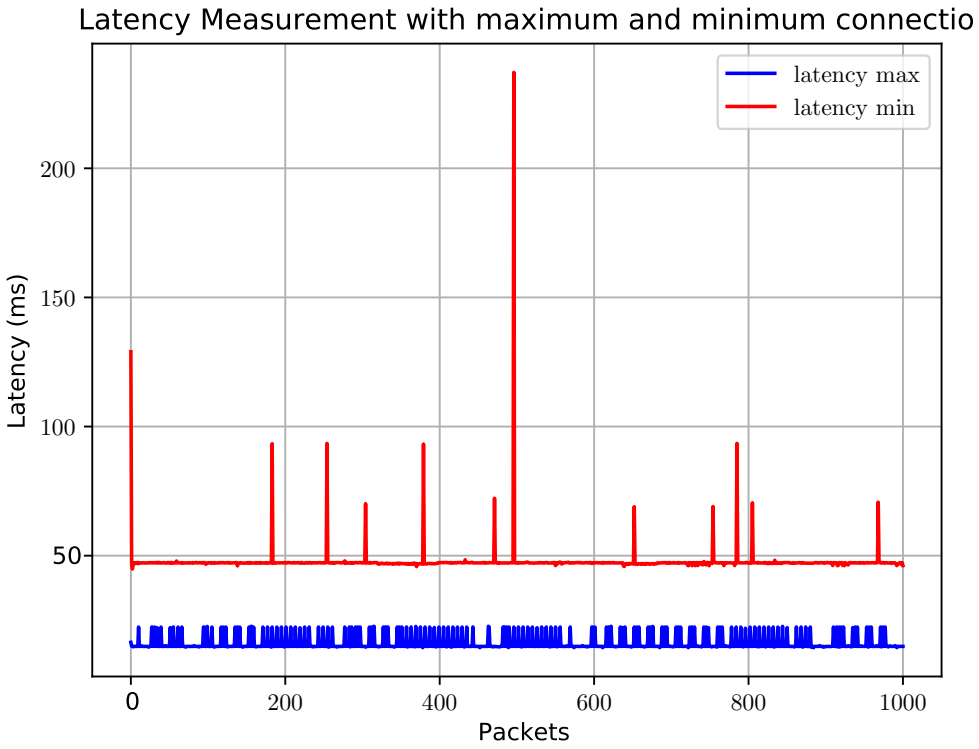

Figure 3.8: Latency with Maximum and Minimum Connection Parameters

3.6 Connection Stability Test

Connection stability test involved 4 nrf52 nodes connected to the raspberry pi. To test the connection stablility a python script was used which based on the given list of bluetooth addresses of all the nodes, connected to them and reconnected in case of a disconnection while maitaining reconnect count in a text file. This test was ran for a full week and resulted in no disconnection events.

4 Comparison of Bluetooth Low Energy Versions

4.1 Introduction

After mentioning the key features and differences between Bluetooth classic and Bluetooth low energy now we would mainly focus on the comparisons of different versions of Bluetooth low energy. Bluetooth low energy is standardized in all smartphones tablets and laptops. It has support in iOS Android Mac OS as well as window 10 and Linux. It is very efficient and ultra-low power protocol. Elaborating the ultra-low power protocol means that when we have very small data packets So our device can sleep as much as possible which basically refers to the sleep mode. The more the sleep mode the device stays in the lower the power consumption which eventually makes it efficient and long lasting .it has a very low RAM footprint. Bluetooth low energy is managed by the Bluetooth SIG(Special Interest Group).This group licensed the Bluetooth low energy technology And it consists of a network of member organizations .SIG It was founded all the way back in September 1998 So it has been active for almost two decades and it is a nonprofit organization. Currently SIG has 36,000 member companies which is quite an impressive number. According to statistics in 2019 there were more than 4.2 billion Bluetooth products shipped. Nordic semiconductor is an associate member in the Bluetooth SIG And is involved with several of the working groups which help develop Different specifications for the products.

4.2 Bluetooth 4.0

- Bluetooth version 4.0 is also known as Bluetooth Smart And it was released on 30th June 2010.The general improvements to Bluetooth version 4.0 that it changed to BLE .
- Bluetooth version 4.0 had the range of 200 feet

- There were several new features that were introduced in Bluetooth Core and the major areas of improvement were changes to BLE modes.
- Enhancement to GATT for low energy which is defined as Generic Attribute Profile
- Security Manager (SM) services with AES encryption(Advanced Encryption Standard)
- Bluetooth 4.1 was a firmware update so it did not had any hardware changes.

4.2.1 GATT

It defines the way that two Bluetooth Low Energy devices transfer data back and forth using concepts called Services and Characteristics. GATT basically defines Format of services and characteristics. It also defines the procedures to interface with the attributes such as discovery read write and notifications.

GATT takes on the same rule as the attribute protocols ATT. These rules are not set per device but rather determined for transaction. It can be explained in such a manner that when the request is sent by the client Then the server responds with the response. In the case of an indication An indication is sent by the server which requires that the Client returns with a confirmation packet.

Similar to the attribute Protocol a device can act as both a GATT and a client at the same time. Now, defining that is what services and characteristics in real are. A service basically is a grouping of one or more attributes and it's meant to group together related attributes that satisfy a specific functionality on the server. Whereas, attribute to within a service can have different types. They could be either characteristics, which hold values or they could be non-characteristic types which help structure the data within that service.

If we take example of a battery service. This kind of service contains one characteristic called the battery level but it also contains other attributes that are not characteristic that helps to structure the data within the service such as the service declaration. A characteristic within a service represents a piece of information or data that the server wants to client. An example could be that the battery level characteristic represents the battery level percentage in a device which can be read by a client and it's contained in the value field of that characteristic.

4.2.2 Characteristics

A characteristic contains other attributes that Help define the value that it holds.

- Properties: (Read, Write, Write without Response, Notify).These properties define how the characteristic value can be used .
- Descriptors: Descriptors are used to contain related information about the characteristic value such as User Description Fields, Enabling Notifications, and a field that defines the presentation of a value such as format and the unit of the value.

4.2.3 Profiles

Profiles are much more in in definition than services and characteristics. They are more concerned in defining the behavior of both the client and the server ,when it comes to services, characteristics, connection and security requirements. Services and their specification on the other hand deal with the implementation of services and characteristics on the slave side only. Now getting into the attribute data operations. There are different types of attribute protocol packets. Attribute Protocol Packets:

- Commands : Sent by the client to the server and no response is required.
- Requests : Sent by the client to the server but in this case the response is required.
- Responses : Sent by the server in response to the request.
- Notifications: Sent by the server to the client to let the client know that a specific characteristic value has changed. In order for this to be triggered the client has to enable the notifications for the specific characteristic of interest. Notification does not require a response from the client to acknowledge received.
- Indications: Indications are very similar to notifications except that they require an acknowledgement to be sent back from the client to let the server know that the indication was successfully received.
- Confirmations: These are sent by the client to the server and those are the acknowledgement packets that are sent back to the server in response to the indication.

4.2.4 Security Manager with AES Encryption

When a couple of devices wish to communicate with each other which at the initial level do not have the security requires the security for the communication to be secure. This process is triggered by a central device such as smartphone which is attempting to access a data value clearly defining the data value in a characteristic on a peripheral device that requires authenticated access. Pairing involves authenticating the identity of two Devices, encrypting that link using a short term key (STK) and then distributing long term keys (LTK) used for encryption. The long term key (LTK) is saved for faster re-connection in the future and this is known as bonding.

4.2.5 Security Modes

The Generic Access Protocol (GAP) defines two security modes, along with several security levels per mode for a BLE connection. Security modes play a vital role in the communication of two devices. The encryption in Bluetooth Low energy is based on 128 bit advanced encryption standard (AES). Long term key is used with an algorithm to create the 128 bit shared secret key. In Bluetooth low energy the Authentication is provided by digitally signing the data using the connection Signature Resolving Key (CSRK). The sending device places a signature after the Data PDU. The receiver verifies the signature using the CSRK. The generic access protocol defined as gab provide two security modes for a BLE connection.

- **Security Mode 1:** This mode enforces security by means of encryption and contains 4 levels.
  - Security level 1: no security in this level no authentication and no encryption is required .
  - Security level 2: in the security level UN authenticated pairing with encryption is required .
  - Security Level 3: Authenticated pairing takes place in the presence of AES-CCM encryption.
  - Security level 4 :Authenticated low energy secure connections pairing with encryption. Level 4 uses elliptic curve Diffie- Hellman P-256 (ECDH) and AES-CCM encryption.
- **Security Mode 2:** Security mode two enforces security by means of sign in data and this also contains 2 levels.

  - Security Level 1: In this security level unauthenticated pairing is done with data signing.
  - Security Level 2 : Authenticated paring is done with data signing.

- **Mixed Security Mode:** When the device is required to support both the security modes Security Mode 1 and Security Mode 2. In mixed security mode it needs to support signed and unsigned data.

4.3 Bluetooth 4.1

In the previous section for Bluetooth version 4.0 we discussed the new features and major areas where improvement was made.Bluetooth Smart technology introduced in Bluetooth 4.0 is a feature within the Bluetooth 4.1 core specification. Bluetooth 4.1 extends the functionality set already provided by Bluetooth 4.0. Bluetooth 4.1 rules up adopted Bluetooth Core Specification Addenda(CSA 1 2 ,3 and 4) while adding new feature and benefits. Bluetooth 4.1 devices can act as both as Hub and the endpoint simultaneously. This is significant in terms that it allows the host device to be cut out of the equation and allows the peripherals to communicate independently which leads to improved data transfer rate. Bluetooth 4.1 offers various key features and benefits which are segmented in the table below in three areas.

Features in Bluetooth version 4.1:

- Improving Usability
- Enabling the Internet of Things (IoT)

In addition to the above listed new features, Bluetooth 4.1 enables the manufacturers to take advantage of important clarifications incorporated into the core specification in the form of errata. Now keeping the compatibility factor under consideration so the devices implementing only the low energy feature will be backward compatible with Bluetooth 4.0 devices that also implement the low energy feature. Devices implementing the Basic Rate/ Extended Data Rate (BR/EDR) core configuration will be backward compatible to all adopted Bluetooth core versions beginning with 1.1 that also implement Bluetooth BR/EDR.

Table 4.1: Link Layer Packet Format

Preamble (1 byte)	Access Address (4 Bytes)	Data Protocol Data Unit (PDU) (2 to 257 bytes in BLE 4.2	CRC 3 bytes

4.4 Bluetooth 4.2

In this section we will focus on the new updates and improvements made in the core specification leading to new features. Bluetooth 4.2 introduces several new features that improve speed and privacy over Bluetooth 4.1 but the main advantage is allowing the chip to use Bluetooth over Internet Protocol version 6 (IPv6) for direct Internet access.

Features in Bluetooth version 4.2:

- LE Data Packet Length Extension
- Link Layer (LL) Privacy
- Low Energy Secure Connection

4.4.1 LE Data Packet Length Extension

The Link Layer (LL) is the part of the BLE protocol stack that takes care of advertising, scanning, creating, and maintaining connections. Each packet consists of four fields: the Preamble, the Access Address, the Protocol Data Unit (PDU), and the Cyclic Redundancy Check (CRC). The packets transmitted during advertising, scanning, or connection creation procedures use the Advertising Channel PDU. The packets transmitted to exchange data with connected devices use the Data Channel PDU.

A Data Channel PDU includes a 16-bit Header, a variable-size Payload field, and an optional Message Integrity Check (MIC) field. In the Bluetooth 4.2 Specification, the maximum size of the Payload field in the Data Channel PDU was increased from 27 bytes to 251 bytes, thus increasing the capacity of the Data Channel by approximately 10 times. In order to have a clear picture the difference between the Data Channel PDU in Bluetooth 4.2 and Bluetooth 4.1 can be illustrated in the tables 4.2 4.3 4.4 4.5:

The Length field in the Header specifies the number of bytes of data following the Header. The size of the Length field in the Header is increased from 5 bits in Bluetooth 4.1 to 8 bits in Bluetooth 4.2, thereby increasing the range of the Length field value

LSB	Data Channel PDU in Link Layer (LL) 4.2	MSB
Preamble (16 bits)	Payload (Maximum of 251 bytes)	MIC (Message Integrity Check) (32bits)

Table 4.2: LE Data Channel PDU in Bluetooth Specification Versions 4.2

Header (16 bits)					
LLID (2 bits)	NESN (1 bit)	SN (1 bit)	MD (1 bit)	RFU (3 bits)	Length (8 bits)

Table 4.3: Header for LE Data Channel PDU in Bluetooth v4.2

LSB		MSB
Header (16 bits)	Payload (Maximum of 27 bytes)	MIC(Message Integrity Check) (32 bits)

Table 4.4: Data Channel PDU in Link Layer v4.1

Distribution of bits in the Header (16 bits)						
LLID (2 bits)	NESN (1 bit)	SN (1 bit)	MD (1 bit)	RFU (3 bits)	Length (5 bits)	RFU (3 bits)

Table 4.5: Header of Data Channel PDU in Link Layer v4.1

from 31 to 255. The Message Integrity Check (MIC) used in encrypted packets is 4 bytes long. Therefore, the maximum possible payload size is 251 bytes in Bluetooth 4.2 and 27 bytes in Bluetooth 4.1. In conclusion, the LE Data Packet Length Extension feature enables applications to get higher throughput and lower power consumption and asymmetric bandwidth .It also enables you to get approximately 2.6 times higher throughput through the Link Layer These benefits are available only when the following conditions are met:

- Both the BLE devices support LE Data Packet Length Extension
- Higher-layer protocols use greater than the default (23 bytes) Maximum Transmission Unit (MTU) size.

4.4.2 Comparison of Payload between Bluetooth 4.1 and Bluetooth 4.2

In Bluetooth 4.1, the payload size is 27 bytes and the total time taken for single transaction is 708 µs that gives the theoretical throughput of 298 kbps. In Bluetooth

4.2, the payload size is 251 bytes and the total time is 2500 µs that gives the theoretical throughput of 784 kbps. This gives approximately 2.6 times throughput for a Bluetooth 4.2 device as compared to a Bluetooth 4.1 device. In Bluetooth 4.1, the 135-byte payload is split into 27-byte payloads and sent over five transactions. In Bluetooth 4.2, the 135-byte payload is sent in a single transaction.

4.4.3 Low Power Consumption

Under the ideal conditions with no air-interference, the LE Data Packet Length Extension feature helps reduce the power consumption due to a more efficient use of the bandwidth. In Bluetooth 4.2, a lower number of transactions are required to transfer a given amount of data compared to Bluetooth 4.1. This reduces the time for which the radio is active and allows the device to remain in a low-power mode for a longer duration, thereby reducing the average current consumption.

4.4.4 Asymmetric Bandwidth

Asymmetric bandwidth means that the TX and RX bandwidths are not same. It is useful in applications like over-the air (OTA) firmware upgrade that require higher bandwidth in the RX direction and lower bandwidth in the TX direction. By a proper selection of values for the local MaxTxOctets and MaxRxOctets parameters, you can achieve asymmetric bandwidth. For example, setting the MaxTxOctets to 251 bytes and the MaxRxOctets to 27 bytes provides more bandwidth in the transmit direction. Similarly, setting the MaxTxOctets to 27 bytes and the MaxRxOctets to 251 bytes provides more bandwidth in the receive direction.

4.4.5 Applications

- Audio-over-BLE can make use of the higher bandwidth to reduce the processing power needed to compress the data.
- Over-the-air (OTA) firmware upgrade can be completed in a shorter time with a lower power consumption
- Internet Protocol Support Profile (IPSP) packets can be exchanged faster, resulting in faster discovery and transactions.
- Data from multiple sensors can be logged faster with the increase in the data payload size.

Slot Availability Mask (SAM)
2 M sym/s PHY for LE
LE Long Range
High Duty Cycle Non-Connectable Advertising
LE Advertising Extensions
LE Channel Selection Algorithm #2

Table 4.6: Features in the Bluetooth Core Specification 5.0

4.5 Bluetooth 5.0

Starting from the evolution of Bluetooth 4.0 which targeted applications that involve connecting sensors and devices with low bandwidth requirements to achieve the longest battery life. Bluetooth 5 takes it even further where it focuses on broadening the range of IoT applications. Bluetooth 5.0 brought us with the features such as twice the speed, four times the range and eight times the advertising capacity of the previous versions of Bluetooth Low energy. In Bluetooth version 5.0 core specifications both BLE and Bluetooth classic are covered so it is important to know which features apply to BLE and which do not. Most of the features introduced in Bluetooth five are focused on Bluetooth Low Energy. The major areas of improvement are mentioned in the table 4.6.

In Bluetooth 5.0 we cannot achieve both the high speed and the long range at the same time. Another important fact is that using the higher speed mode would allow you to lower the power consumption of your device since the radio is on for shorter periods of time. Where as using the longer range mode the power consumption will go up and the data rat will be slower. Following are the highlighted features of Bluetooth 5.0:

- 2x Speed due to 2M PHY
- 4x Range due to Coded PHY
- 8x Advertising Capacity

Now let us find out how double the speed on four times the range is achieved in Bluetooth 5.0. It does so by the addition of two new physical layers or the radio component of the device. Originally we had the one megabit physical layer(1M PHY) which existed in BLE versions 4.0 up to 4.2. However, in Bluetooth 5 we have the 2M PHY which achieves the higher speed of up to 2 Megabits per second data rate and the Coded PHY which achieves a longer range up to 4 times the original range using

Forward Error Correction technique. 8x advertising capacity is also one of the new feature and this enabled by advertising extensions. The advertising packets to be sent on the secondary advertisement channels which are the same as the data channels. The difference between Extended advertisements and advertisements sent on the primary channel is that they allow larger payloads 255 bytes verses is 37 bytes .These advertising extensions could be sent on any PHY (2M, 1M, Coded).Primary advertising packets can only be sent on 1M (1 Megabit PHY) or the Coded PHY. A new mode in Bluetooth 5.0 is a periodic advertising mode which is a special case of extended advertisements. It allows a scanner device to be synchronized to a broadcaster reading its advertisement data reading periodically without a connection. This has the protention of lowering the power consumption on the scanner side since the radio only needs to wake up certain times.

IoT Applications:

- Low-quality video streaming over short distances
- Long-distance remote control applications
- Broadcast of additional information from beacon devices
- Synchronized monitoring of sensor data without the need for a connection.

4.6 Bluetooth 5.1

The new feature introduced in Bluetooth v5.1 are listed in the table 4.7

Direction Finding
GATT caching
Advertising Channel Index
Periodic Advertising synchronization Transfer

Table 4.7: Features in the Bluetooth Core Specification 5.1

4.6.1 GATT Caching

Another feature introduced in Bluetooth version5.1 is GATT Caching. This is defined as the ability to store and cache the attribute table of a GAT server. This allows for lower power consumption since the radio is on for shorter periods of time and for faster connectivity as well. Before Bluetooth version 5.1 this was restricted do bonded devices but now in Bluetooth version 5.1 it is allowed for unbonded devices as well.

4.6.2 Use of Advertising Channel Indices

This is the third category of enhancements related to Bluetooth advertisements. This includes two new features. Primary advertising occurs on channels 37,38and 39(or a subset of these channels). Before version 5.1 We were required to utilize these channels in that specific order. However, starting with version 5.1 we can now use these channels in any order that we want. This helps primarily for large-scale Bluetooth mesh deployments since devices in such a network utilize the advertising packets for communication. For instance, randomizing the order of channels used could reduce advertising packet collisions.

4.6.3 Periodic Advertising Sync Transfer

This feature of Bluetooth version 5.1 is also related to advertising. This is a mode that utilizes extended advertisements to allow a device to sync to speak to another continuously advertising device and receive this data As it changes without having To establish a connection. In the first stage of this mode the listening device needs to acquire the sync Confirmation to be able to follow the advertising packets continuously. In Bluetooth version 5.1 a new feature allows a device that previously synced to an advertising device to transfer the sync information to a third device that wants to sync the advertising device as well. This feature is targeted primarily at the future Audio over LE standard for devices such as hearing aids.

4.7 Bluetooth 5.2

The new features introduced in Bluetooth version 5.2 are listed below.

- Enhanced Attribute Protocol
- LE Power Control
- Isochronous Channels
- LE Audio

4.7.1 Enhanced Attribute Protocol

Attribute Protocol is the upgraded version of the original Attribute Protocol (ATT) whereas the original unenhanced Attribute Protocol operates in a sequential manner.

The enhanced attribute protocol provides a way to perform concurrent or parallel transactions between a BLE client and a server and potentially reduces the latency operation in some applications. For instance, this is useful on a smartphone where multiple applications may be interfacing with a Bluetooth low energy device But by utilizing the enhanced attribute protocol an applications attribute Transaction would not be blocked While another application's attribute transaction is currently in progress ,Essentially allowing different applications to interact with a Bluetooth low energy device in parallel. The Enhanced Attribute Protocol is optional per the specification and it also requires encryption of the connection between two Bluetooth low energy devices which makes it inherently more secure than the original unenhanced attribute protocol.

4.7.2 LE Power Control

In wireless communication Received Signal Strength Indicator (RSSI) can be used to estimate the distance of the receiver from a transmitter if the original transmit power is known to the receiver. Wireless receivers have an optimal received signal strength range. Higher or lower than this range may cause issues with decoding the received signal so the Received Signal Strength Indicator (RSSI) within this range provides better signal quality . With a new LE power control feature a receiving device monitoring the level of the signal, the (RSSI) from a connected device may request a change in the transmit power level used by its peer either direction. A transmitter may also change the transmit power voluntarily and relay that information to the receiver. Utilizing LE power control and Keeping the RSSI within the optimal range after receiver provides a few benefits. Some of those benefits include better control over signal quality of the signal reducing error rates at the receiving end improving co-existence with other signal in 2.4Ghz band including those other than Bluetooth such as Wi-Fi and Zigbee. However, support for this feature is optional but if the two devices support the feature then they must use it for power management control .

4.7.3 Isochronous Channels

Isochronous means occurring at the same time .In this context it means to supporting data transmission that are time-sensitive And providing support for a synchronized Rendering of the data across multiple receivers. This new feature serves as the foundation For the next generation of Bluetooth audio refer to as LE Audio. It introduces a new physical channel in Bluetooth low energy known as the LE Isochronous (ISOC)

physical channel which can be used on any of the LE PHY'S (Physical Layer) including 1M,2M and LE Coded PHY. Isochronous channels are supported for both connection oriented and connectionless communication such as broadcasts. In connection oriented communication each stream is referred to as a Connected Isochronous Stream (CIS). When Connected Isochronous Stream (CIS) needs to be synchronized such as one's sent to the left and right earbud, they are configured to be part of a single group referred to as a connected Isochronous Group (CIG).Streams that are part of the same CIG share timing reference data which is necessary for synchronized stream rendering at the multiple receivers CIG's allow bi-directional Data transfer Such as in the ear buds that contain microphones and for sending control data to this source device. Device may create multiple connected isochronous Groups for connectionless communication such as broadcasts. A group of streams may be used to stream data from a single source to multiple syncs. Each stream is referred to as a broadcast Isochronous stream Isochronous Stream (BIS).A group of BIS are referred to as Broadcast Isochronous Group(BIG). An example for this would be a Television streaming audio data to multiple sync such as different individuals wearing earbuds. Just as in the case of Connected Isochronous Group (CIG) a device may also create multiple Broadcast Isochronous Groups (BIG's).

One important parameter in Isochronous channels is the ISO interval. It defines the interval at which the events occur .Each even is split into multiple sub events. The ISO interval ranges from milli seconds to 4 milli seconds. In connection oriented communication each sub event the master will send a packet to the slave and the slave will respond with a packet. However, in connectionless communication only the master will send a packet in each event. In this case the packets could be either isochronous data or the broadcast control information data. Isochronous channels also support data retransmissions. However, they differ between a connection oriented and connectionless communication. In the case of broadcast Isochronous streams retransmissions are sent by the master without influence from the slave or the slaves. In the case of Connected Isochronous Streams (CIS) retransmissions are sent when a slave has not acknowledged a packet. In order to reduce the risk of packet loss retransmissions are sent on different channels than the original.

4.7.4 LE Audio

Isochronous channels are the foundation of the for this new feature. LE audio operates on Bluetooth Low energy rather than the traditional Bluetooth Classic or BR,

EDR(Basic Rate, Extended Data Rate). LE Audio not only provides the same features as Bluetooth Classic but also introduces a few new enhancements such as new higher quality and more efficient Codec known as LC3. It also introduces multi stream capabilities such as transmitting separately, left and right audio streams and even streams in multiple languages. Other feature includes support for enhanced Bluetooth hearing aids and support for audio sharing and audio broadcasting.

5 Comparison of Bluetooth, BLE and Bluetooth Mesh

5.1 Classic Bluetooth and Bluetooth Low Energy

Bluetooth low energy version 4.0 which features low energy technology was launched in 2011 where as classic Bluetooth is also the part of version 4.0But more typically referred to the instance version 2.1 Plus EDR v4.. The important note is that both the flavors Bluetooth low energy and classic Bluetooth will be used but they will be used differently .Classic Bluetooth technology is typically the preferred choice as it achieves substantially greater throughput than Bluetooth low energy technology. In classic Bluetooth technology we have standard Bluetooth profiles That dictate the functionality and we also have the possibility for up Seven simultaneous connections Bluetooth.

Bluetooth low energy technology enables new applications and is ideal for applications Requiring episodic or periodic transfer of small amounts of data. Bluetooth low energy also features ultra-low power consumption, quick connection times, large number of circle slaves And the advertising functionality which makes It possible for a slave to announce that he has something to transmit to other devices that are scanning. A glimpse of the features of both the technologies can be compared in the table 5.1.

Now, comparing the difference in both the technologies by digging a little deeper so, we analyses that Classic Bluetooth and Bluetooth low energy both have their strengths and weaknesses. For instance data transfer rates with classic Bluetooth using Enhanced Data Rate (EDR) Bluetooth version 2.1+EDR can exceed 2Mbps actual payload But practical transfer rates for Bluetooth low energy are below 100Kbps.Many robust features of classic Bluetooth technology Are inherited in Bluetooth low energy technology including Adaptive Frequency Hopping as well as part of the L2CAP interface. Just as with classic Bluetooth low energy Is based on a master connected to number of slaves. However, in Bluetooth low energy the number of slaves Can be very large. How large depends on the implementation And available memory. In Bluetooth

Classic Bluetooth	Bluetooth Low Energy
V2.1+EDR/ v4.0	Version 4.0
Streaming Data	Low data transfer rates
Higher Transfer Rate	Ultra-low power consumption
Standard Bluetooth profiles SBB, DUN, BAN	Connection times: few milliseconds
It can handle up to 7 Slaves.	Profiles based on the (GAP) large number of slaves Advertising Functionality

Table 5.1: Comparison of Classic Bluetooth and Bluetooth Low Energy

low energy technology The actual connection times are of only a few milliseconds And thereby the connection is quickly Initiated as a device wakes up. A Bluetooth low energy device has a very low power consumption Since it is in sleep mode most of the time. Smartphones and tablets are widely used in everyday life by installing a tailored application. The everyday smartphone or tablet becomes a powerful and cost efficient tool for industrial and medical applications. For instance an application can be designed to gather certain data Or to perform specific tasks such as to act as an HMI panel or a remote control. Classic Bluetooth implementations are single mode implementations but with the addition of Bluetooth low energy there are also single mode Bluetooth low energy devices.

The two technologies are fundamentally from an application perspective. One either implements single mode Or dual mode solutions depending on required application's need. Classic Bluetooth implementations are used in Bluetooth applications where streaming data is used.

Single mode Devices: Single mode Bluetooth low energy devices are also known as Bluetooth smart devices and they are optimized for small battery operated devices with a low cost and a low power consumption at focus.

- Classic Bluetooth OR Bluetooth low energy
- Application: Heart rate sensor

Dual-Mode Devices: Bluetooth dual mode devices are also known as Bluetooth smart ready devices As include both, Bluetooth Low energy and classic Bluetooth. A typical dual mode device is a smart phone or a PC

- Classic Bluetooth AND Bluetooth low energy
- Application: Smart phone, PC.

5.1.1 Bluetooth Technology Differences

Parameters	Classic Bluetooth Technology	Bluetooth Low Energy
Data Payload Throughput	2Mbps	Nearly 100kbps
Robustness	Strong	Strong
Range	up to 1000 Meters	up to 250 Meters
Large Scale Network	Weak	Good
Low Latency	Strong	Strong
Connection Set up Speed	Weak	Strong
Power Consumption	Good	Very Strong
Cost	Good	Strong

Table 5.2: Parameter Comparison of Classic Bluetooth and Bluetooth Low Energy

As mentioned earlier classic Bluetooth has a determined set of profiles that defines the application behaviors that Bluetooth devices use to communicate with each other. Bluetooth low energy technology profiles are different from those used in classic Bluetooth. Unlike classic Bluetooth the product developers can develop their own profiles and services.

5.2 Bluetooth Mesh

One of the most exciting updates from Bluetooth Technology is the Bluetooth mesh Network. In this section our main focus would be on the basic terminologies in Bluetooth mesh network ,the purpose of and the main advantages of Bluetooth mesh network. Bluetooth Mesh was released in July 2017. The basic goal of Bluetooth mesh was to increase or extend the range of Bluetooth networks. The other major purpose was to support as many Industrial Level application as possible. Earlier versions of Bluetooth supported two different topologies.

- One to One (When two BLE devices are connected to each other)
- One to Many (BLE device being in the broadcast such as beacons)

But with the arrival of Bluetooth mesh a new topology is introduced where devices can now operate in a many to many topology which is known as mesh. In a normal mesh network devices can set up connections with multiple other devices within the network. Bluetooth mesh builds on top of BLE. It specifically utilizes the advertising state of BLE devices. An interesting thing to know here is that devices within a mesh network do not connect to each other like traditional BLE devices do. Instead, they

use advertising and scanning states in order to relay messages to each other. The Bluetooth mesh standard defines how different devices operate within the network based on their roles. Different devices here refers to both type of the devices, the one with live power and the other being the low power consuming battery powered devices. If we discuss certain basics of Bluetooth mesh so it is such a technology which supports all versions of Bluetooth low energy which means going back up to Bluetooth version 4.0. However it is a separate specification and standard that is not part of the official Bluetooth specification and this includes Bluetooth 5 as well.In order to have a better understanding of Bluetooth mesh I would define the basic terminologies used in Bluetooth mesh.

- Node : The devices that are part of a Bluetooth mesh network are known as Nodes.
- Unprovisioned Device: These are the devices that are not part of the Bluetooth mesh network.
- Elements
- States
- Properties
- State Transitions
- Messages
- Addresses
- Model
- Scene

As soon the Unprovisioned devices get the provision, it joins the network and also become a node. A node may contain multiple parts which can be controlled independently and these individual parts are known as elements. A nice example for this can be a light fixture contains multiple light bulbs. These light bulbs can be switched on or switched off independently. The states can be easily understood by the concept of a bulb as an example being switched on state or a switched off state. Properties are the parameter which add context to the state. An application is a temperature sensor(out door in comparison to indoor temperature). State transition defines the state change

from one state to another. These state transition can occur instantaneously or also over the period of time. The nodes within a mesh network send messages to each other. These messages are used to control a node or relay information between nodes or to report status to each other. The type of message is defined through a unique operation codes. In regards to the messages there are two categories. The first being the Acknowledged messages that require a response from the receiver node or nodes and the other being the Unacknowledged messages which do not require any response. The purpose of responses is to allow confirmation of the receipt of the message and also sending back data related to the message that was received.

Messages are of three types:

- Get Messages: They request the state from a node
- Set Messages: These messages change the value of a given state.
- Status Messages: These messages serve as responses to a Get message, a Set message

Address identify the source and destination of a message .The messages must be sent to and from the address. In Bluetooth mesh there are three types of addresses:

- Unicast Address: It uniquely identifies a single node and is assigned during and is assigned during a process which is known as provisioning process.
- Group Address: This address identifies a group of nodes. Group addresses could also be of two categories: The first being the SIG fixed which are defined by the Bluetooth SIG and the second being the dynamic addresses which are defined by the user via an application through which it can be configured. Group addresses in general reflect a physical grouping of nodes such as all nodes within a room in a house.
- Virtual Address : This address may be assigned to one or more elements (one or more nodes) The benefit of using group or virtual address is that adding or removing nodes does not require reconfiguration of the other nodes.

The benefit of these models is that these models can be extended to include additional functionality instead of modifying the original model but there are models which cannot be extended and they are defined as root models. Models are also divided into three categories:

- Server Model: In this model is a collection of states such as state transitions, state bindings and message which an element containing the model may send or receive.
- Client Model : Client model does not define any state but it focuses on the messages such the Get , Set and Status messages sent to a server Model.
- Control Model: This Model contains both, the Server and a Client Model allowing communication with other server and client Models.

5.2.1 Scenes

A scene is defined as a stored collection of States. It is identified by 16 bit number which is unique within a mesh network . A scene allows activating of one action to set multiple states of different nodes in the network. They can be generated on demand or scheduled at a specific time. In order understand a scene an example could be given. A scene may be configured in such way that that the user wants to set the room temperature of a certain a certain room and along with that also wants to configure the closing of window blinds. Just imagine all the states being triggered or activated by just triggering one action would be a lot more exciting.

5.2.2 Types of Nodes in Bluetooth Mesh

All the nodes in a Bluetooth mesh network can send and receive mesh messages In mesh network there are certain optional feature that gives node special capabilities. There can be many different nodes in a mesh network.

- Relay Node: it is the node that can retransmit messages that are broadcast from other nodes allowing them to reach the network and extend the reach of the network.
- Low Power Node: A Low Power Node (LPN) is a power constraint node. These nodes needs to be in sleep mode as often as possible and turn off the radio most of the times. Low Power Nodes (LPN) are mostly concerned with sending messages such as in sensor devices that read data from the environment and send them to the network.
- Friend Node: Friend node is not a power constraint. It needs to keep the radio on all the time to be able to listen to the broadcast messages in order to relay those

messages to the Low Power Nodes. As we know that the Low Power Nodes are not awake all the time so the Friend Node needs to cache those messages for the Low Power Nodes and when the Low Power Node wakes up from the sleep mode and pulls the Friend Node and that is when the Low Power Node can receive those messages.

- Proxy Node: A node supporting proxy feature is able to relay/forward messages between non-mesh Bluetooth devices and a mesh network. [HPG+20]

The benefits of a mesh network are:

- Extended range: Since nodes can relay messages to far away nodes via the nodes in between them, this allows a network to extend its range and expand the reach of devices.
- Self-healing capabilities(Resiliency): Self-healing refers to the fact that there is no single point of failure. If a node drops from the mesh network, the other nodes can still participate and send messages to one another. However, this is only partially true for Bluetooth mesh since it has different types of nodes within the network, some of which other nodes may depend on.
- Scalability and reliability: An import aspect for the mesh network is scalable. It allows very easy addition of devices

5.3 Working Process

5.3.1 Provisioning Process

Provisioning is the process of adding devices to the mesh network. As mentioned in earlier section that a device that a device that gets added to mesh network is called a node. The provisioner is usually a tablet or a smartphone but the provisioning process involves five steps:

5.3.2 Beaconing

In beaconing the Unprovisioned device announces its availability to be provisioned by sending the mesh beacon advertisements in the advertisement packets. This process may be activated by a defined sequence of button presses initiated by the user on the Unprovisioned device. When the provisioner discovers the Unprovisioned device,

it send an invitation to this device. This invitation uses a type of PDU called the provisioning Invite PDU. The Unprovisioned device then responds with information about itself in Provisioning capabilities PDU. Included in this PDU are the number of elements this device supports, the set of security algorithms supported, the availability of its public key using an auto band technology, the availability for this device to output the value to the user as well as the ability for the device to allow a value to be input by the user.

5.3.3 Security Feature

Security feature in Bluetooth mesh makes use of combination of symmetric and asymmetric key. Public keys are exchanged between the provisioner and the device to be provisioned. This is either done directly or through an auto band channel. The final step is to authenticate the Unprovisioned device. Authentication requires action by the user by interacting with the both the provisioner and the Unprovisioned device. Authentication depends on the capabilities of both device used. In one case called the output auto band , the Unprovisioned device could output a random sing or multiple digit numbers to the user in some form such as blinking an LED a number of times The authentication step also uses a confirmation value generation and a confirmation check. After authentication is complete each device derives a session key, using the private key and the public key set to it from the other device. The session key is then used to secure the connection for exchange of additional provisioning data. This data includes a network key , a device key, a security parameter (IV Index) and a Unicast address which is allocated by the provisioner. Now the Unprovisioned device becomes what is known as a node.

In regards to security, in Bluetooth mesh all the messages are encrypted and authenticated. Network security, device security and application security are all handled separately in Bluetooth mesh network. Security key can be changed during the life of a network.

5.3.4 Network Key

The procession of this shared key makes the device a part of the network also called a node. There are two keys derived from the Network kay which includes the network encryption key and the privacy key .The procession of the Network key allows a node to decrypt and authenticate up to the network layer , allowing relaying of messages but not the application data decryption.

5.3.5 Application Key

It is a key shared between a subset of nodes within a mesh network .Normally for those that participate in a common application. An example could be a lightning system key application key would be shared between light switches and light bulbs but not with a motion sensor or a thermostat. This application key is use to decryption and authenticating messages. Application keys are valid within one mesh network and not across the other networks.

5.3.6 Device Key

It is a device specific key that is used during provisioning for securing communication between the device which is the node and the provisioner during the provisioning process. The major security concern with the mesh network is gaining access to a network through a detached device that used to be part of the network.

5.3.7 Applications of Bluetooth Mesh Network

One of the most popular usage of bluetooth mesh is in Commercial Lighting Solutions and Sensor Network Solutions across several markets.

Anhang

Bibliography

[Afa18] Afaneh, Mohammad: *Intro to Bluetooth Low Energy.* Novel Bits LLC, 2018

[Afa20] Afaneh, Mohammad: *Bluetooth 5 speed: How to achieve maximum throughput for your BLE application.* `https://www.novelbits.io/bluetooth-5-speed-maximum-throughput`. Version: Jun 2020

[ble] *Connection timing with Connection Event Length Extension.* `https://infocenter.nordicsemi.com/topic/sds_s140/SDS/s1xx/multilink_scheduling/extend_connection_event.html`

[BSCN12] Bormann, Carsten ; Shelby, Zach ; Chakrabarti, Samita ; Nordmark, Erik: *Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs).* RFC 6775. `http://dx.doi.org/10.17487/RFC6775`. Version: November 2012 (Request for Comments)

[CCG18] Campos, J. ; Colteryahn, S. ; Gagneja, K.: IPv6 transmission over BLE Using Raspberry PI 3. In: *2018 International Conference on Computing, Networking and Communications (ICNC)*, 2018, S. 200–204

[Com06] Commons, Wikimedia: *UDP Header.* `https://commons.wikimedia.org/wiki/File:Header_of_UDP.jpg`. Version: 2006

[Com11] Commons, Wikimedia: *IPv6 Header.* `https://commons.wikimedia.org/wiki/File:IPv6_Header.svg`. Version: 2011

[goo] *IPv6.* `https://www.google.com/intl/en/ipv6/statistics.html`

[HD98] Hinden, Bob ; Deering, Dr. Steve E.: *Internet Protocol, Version 6 (IPv6) Specification.* RFC 2460. `http://dx.doi.org/10.17487/RFC2460`. Version: Dezember 1998 (Request for Comments)

[HPG+20] Hernández-Solana, á. ; Pérez-Díaz-De-Cerio, D. ; García-Lozano, M. ; Bardají, A. V. ; Valenzuela, J.: Bluetooth Mesh Analysis, Issues, and Challenges. In: *IEEE Access* 8 (2020), S. 53784–53800. http://dx.doi.org/10.1109/ACCESS.2020.2980795. – DOI 10.1109/ACCESS.2020.2980795

[ipv] *Layers and Protocols*. https://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.iotsdk.v0.9.0%2Fiot_layers.html

[NJT07] Narten, Dr. T. ; Jinmei, Tatsuya ; Thomson, Dr. S.: *IPv6 Stateless Address Autoconfiguration*. RFC 4862. http://dx.doi.org/10.17487/RFC4862. Version: September 2007 (Request for Comments)

[NSI+15] Nieminen, Johanna ; Savolainen, Teemu ; Isomaki, Markus ; Patil, Basavaraj ; Shelby, Zach ; Gomez, Carles: *IPv6 over BLUETOOTH(R) Low Energy*. RFC 7668. http://dx.doi.org/10.17487/RFC7668. Version: Oktober 2015 (Request for Comments)

[SIG19] SIG, Bluetooth: Bluetooth Market Update. Version: 2019. https://www.bluetooth.com/wp-content/uploads/2018/04/2019-Bluetooth-Market-Update.pdf. 2019. – Forschungsbericht

[spe13] *Bluetooth Core Specification Version 4.0 [Vol 0]*. https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=456433. Version: 2013

List of Figures

List of Tables

Declaration

The present thesis is a group work. Mr. Umer Mahmood was responsible for developing and implementing the experiments. He worked on chapters 2 and 3. Mr. Qasim Ali was responsible for litrature review and investigation of technologies being used. He worked on chapters 1, 4 and 5. We declare that the work is entirely our own and was produced with no assistance from third parties. We certify that the work has not been submitted in the same or any similar form for assessment to any other examining body and all references, direct and indirect, are indicated as such and have been cited accordingly.

Ilmenau, den 31. 12. 2011 Umer Mahmood, Qasim Ali