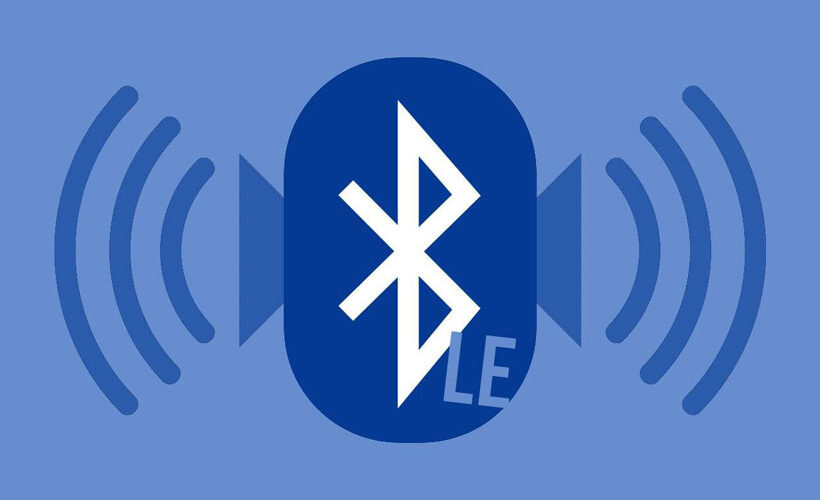
Communication protocols for IoT devices

Wireless Sensors Networks

Bluetooth Low Energy Overview



Presented by :

* *Amour Chadi*
* *Amour Redouane*
* *Meslouh Mohamed Arselan*
* *Meneceur Leila*

5 ISS / MSIoT





**Table of Contents**

**Introduction 2**

**1.General concepts 2**

**2. Physical layer 4**

**3. Data Link layer 7**

**4. Security mechanisms 15**

**5. Power consumption 20**

**Conclusion 22**

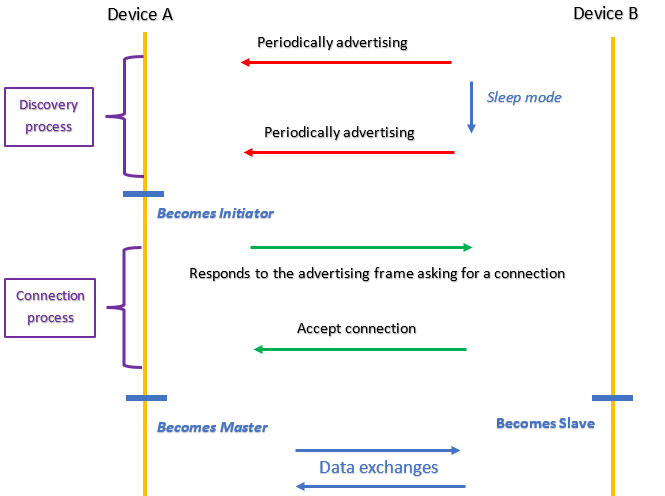
**Bibliography 23I**

**Introduction**

Bluetooth Low Energy (BLE) also known as Bluetooth Smart is the 4th version of Bluetooth standard released in 2010. This wireless type of communication, approved by the Bluetooth Special Interest Group (SIG), aims to become the best alternative to the huge number of standard wireless technologies already existing and widespread on the market (IEEE 802.11b (Wi-Fi), ZigBee, etc.). Overall, BLE stands out from Bluetooth due to its low energy consumption. It has a maximum scope of **100 meters**. The synergy between good performance and ubiquitous diffusion (today, BLE is available in all PCs, tablets and smartphones) makes BLE an excellent candidate for a great variety of applications. This report will first tackle the fundamental concepts of BLE. Once we have got in mind those concepts, we will go deeper by understanding some core layers : the Physical layer, the Link layers enabling data exchanges and a little insight of the security mechanisms involved.

**1. General concepts**

Let us now introduce some of the main BLE concepts so we can next focus on how it works physically (Phy Layer) and logically speaking (MAC Layer). BLE is based on a communication following the scheme below :



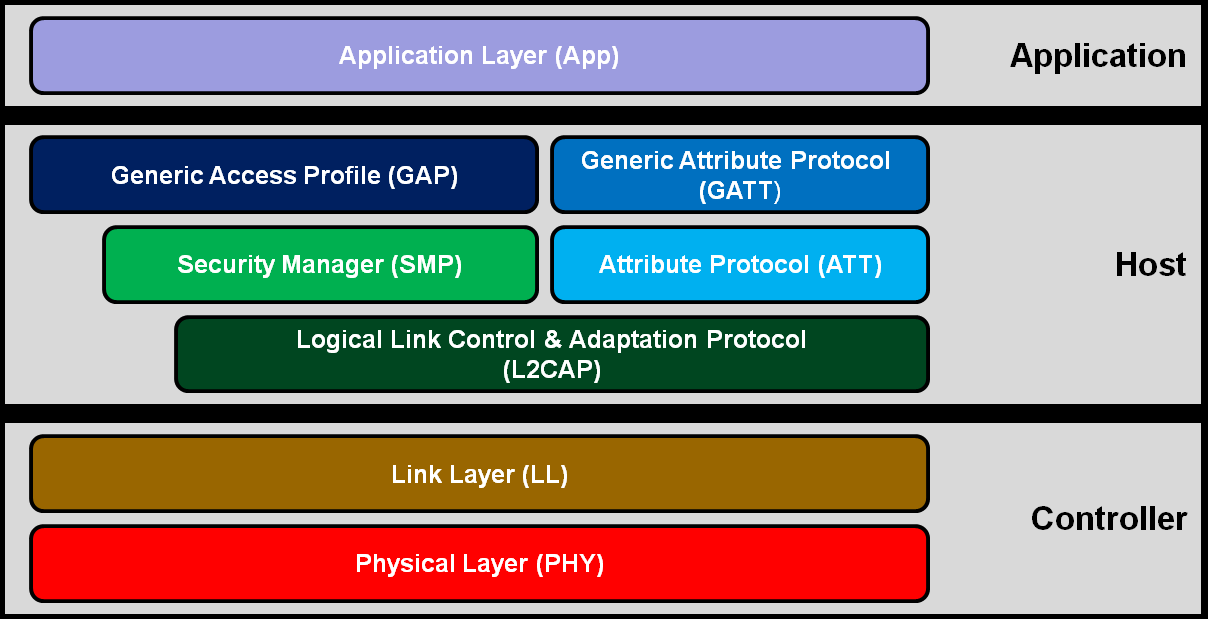
As shown in the above picture, BLE is based on a Master/Slave communication. Obviously some steps are missing here but we are only dealing with BLE core behavior. At first, device B is constantly and periodically diffusing packets to make other devices aware of its presence. Between each packet, device B is in a sleep mode enabling energy saving. Of course, this does not entirely justify the “Low Energy” aspect of BLE communications. Meanwhile, device A is scanning the environment which means listening to advertising packets. Once device A has detected device B packets, device A sends a connection request. After device B accepts this connection, it becomes a Master/Slave communication where Master leads the communication exchange. This example depicts how a BLE exchange occurs.

Thus the communication takes 3 steps :

* advertising / scanning
* connecting / accepting connections
* exchanging data

Besides all, there is a more sophisticated architecture under it making these different steps possible, as shown in the picture below :

BLE global architecture can be depicted as below :



**The controller container**

* Physical Layer refers to the medium of communication used (i.e. waves) to vehicle data packets with some added bytes (frequency, gain etc.). The Controller has a dedicated radio peripheral used in data transmission and data reception (TX and RX registers).
* Link Layer has a major role controlling medium access to communicate from a device to another. That is to say establish some medium configurations to assure frames transport
* What is important to notice is that the Controller device will be responsible for these two layers and it will also be the interface between the Link Layer and L2CAP sublayer. We will see it in part 3

**The Host container**

* The host container has a major role in logical analysis and security mechanisms. That means understand/send the core data from/to the other device which requires specific known headers that we will see in part 4.

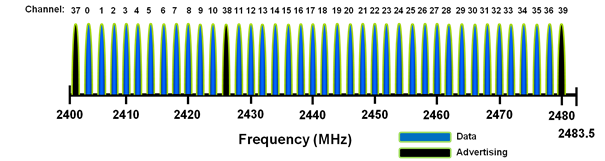
We can now understand how important are those different layers that will be our main subject discussion for the next 3 parts.

**2. Physical Layer**

**2.1. Frequency Bands and channel arrangement**

Bluetooth low energy operates in the 2.4 GHz ISM (Industrial Scientific Medical) band (2402 MHz - 2480 MHz), which is license-free in most countries. The band is divided into 40 channels with 2MHz spacing.

Three of the 40 channels (37,38,39) are advertising channels, used for device discovery, connection establishment, and broadcast while the other 37 channels (0 to 36) are Data channels.

For instance, if we want to get some heart rate data from a smartwatch on our smartphone, the frequency used will belong to the set {0, ... ,36}.

**2.2. Adaptive Frequency Hopping**

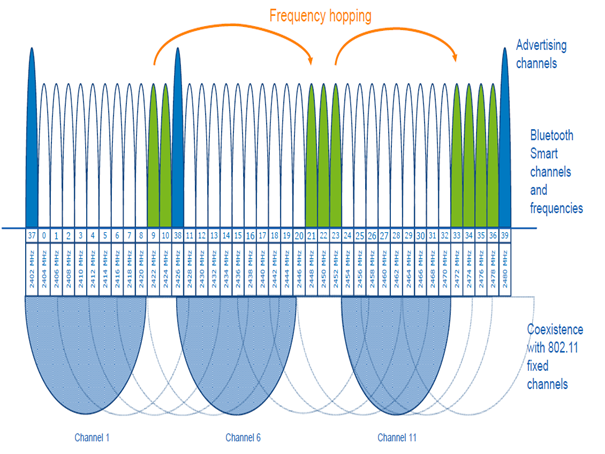
In data use, to be able to connect to the Master device, a frequency channel among the 37 data channels is given to the Slave device. Each time that frequency changes so it doesn’t overstep on an already used frequency channel.

For that purpose, a frequency hopping algorithm is used to cycle through the 37 channels.

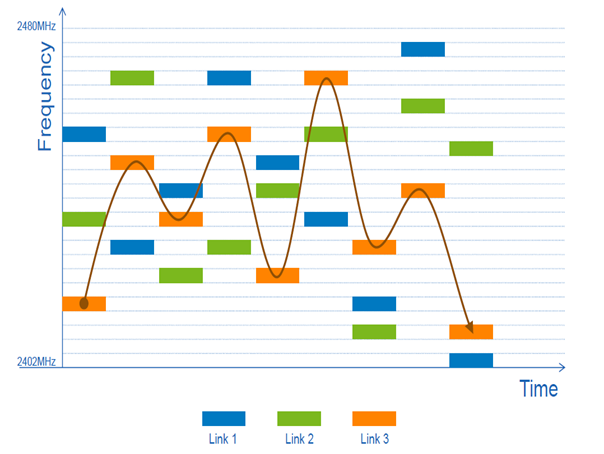
*fn+1=(fn+hop)mod 37*

Where *f*n+1 is the frequency to use on the next connection event and *hop* is a value that can range from 5-16 and is set when the connection is created. It is added onto the last frequency module 37.

This mechanism is used by the link layer to remap a given packet from a known bad channel to a known good channel so that interference from other devices (i.e., Wi-Fi) is reduced.

Suppose, for example, that a BLE device is in the same area as several Wi-Fi networks on channels 1, 6, and 11. The BLE device would mark channels 0-8, 11-20, and 24-32 as bad channels. This means that when the two devices are communicating, they would cycle through the channels and remap these channels to a set of good channels as shown:

The following diagram shows three active BLE connections, showing the frequency hopping sequence (frequency hopping on Link 3 is outlined for clarity).

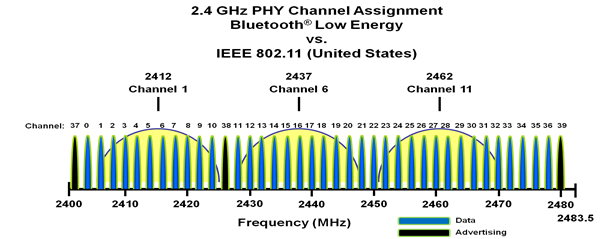


The diagram shows that frequency hopping provides a robust method for maintaining a connection in the presence of interference or many other devices in the radio range.

**2.3. Coexistence with 802.11 Wi-Fi**

The BLE shares the same frequency Band with the 802.11/ Wi-Fi.

Thus, the advertising channels have been assigned center frequencies that minimize overlapping with the Wi-Fi channels.



**2.4. Modulation and transmission power**

All the physical channels of BLE use Gaussian Frequency-Shift Keying (GFSK), whereby the data pulses are filtered with a Gaussian filter before being applied to alter the carrier frequency, in order to make the frequency transitions smoother. The modulation index of 0.5 allows reduced peak power consumption.

In Bluetooth 4.0, 4.1, and 4.2 specification, the physical layer data rate is 1 Mbps with 1 bit per symbol.

The Bluetooth 5 standard introduces an additional 2M PHY rate for faster throughput or shorter TX and RX times.

The recent changes in the Bluetooth and regulatory standards allow Bluetooth Smart devices to transmit up to 100 mW (20 dBm) transmit power. However not all countries allow the 100 mW transmission power to be used because Bluetooth low energy radio can drop down to two RF channels when there is significant interference.



**3. Data Link Layer (MAC + LLC Sublayers)**

In this part, we will try to explain how the Data Link Layer enables us to expose services to the Host Layer. The Link Layer stands on MAC layer protocol ensuring frames exchanging by taking control of the Physical layer . The actual Data Link Layer is composed of the Link Layer and L2CAP sublayer. But since the L2CAP sublayer is part of the Host layer, we will be more focused on the Link Layer role (MAC role)

**3.1. Core concepts**

Let us remind that there are only two types of frames between BLE devices :

advertising frames and data frames. The first one is used to discover devices and establish connections and the second one is used to exchange ressources (like music, heart rate etc.)

We can ask ourselves how a device would know which one of the following cases we are in : either advertising or data communication ?

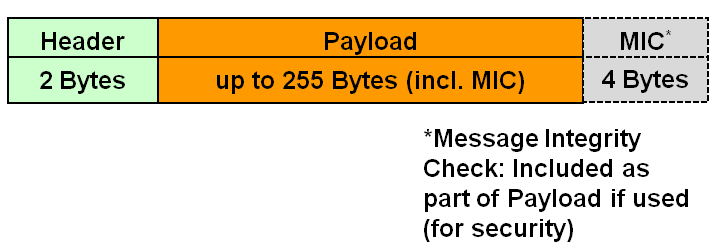
As seen in part 2, the answer stands on the frequency (channel) used, there are only 3 frequency channels (among the 40 channels) available for advertising packets. This is where Data Link layer plays a major role :

* controlling the communication channel by defining the frequency algorithm so both Master and Slave can identify the type of frame among the many frames that are exchanged in the network
* Setting connection intervals (TDMA)
* addressing the frame to the right device using devices addresses (MAC addresses)

We can then understand more precisely how the Data Link layer proceeds.

The BLE Link Layer has only one packet format used for both advertising channel packets and data channel packets :

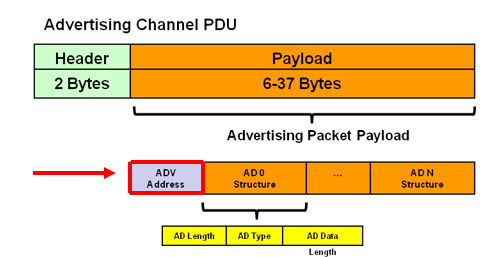
# 

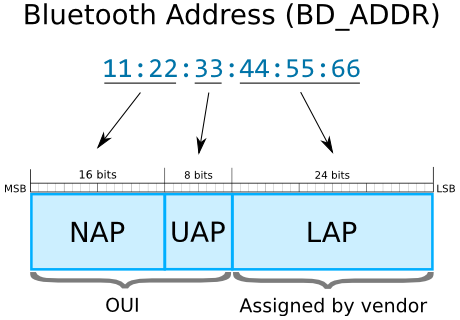


**3.2. Discovery Process**

This is the first step we saw to establish a BLE communication. Here we are dealing with advertising packets. There are actually different advertising packet types but it is not really relevant to review them all. We only have to understand that the 2 bytes of the Link Layer Header will code the type of advertising.

The Link Layer payload is really important though, especially in the Discovery process. This payload will contain Bluetooth Device Addresses (similar to MAC addresses) that enable frames reaching the targeted device :



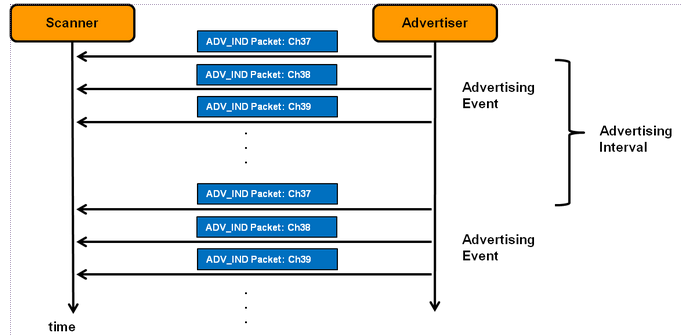


The picture below gives an example of what a MAC device address looks like.

Once the packet is all set, the Link Layer packet is transmitted (by the Controller) to

the Physical Layer if it is available. That means using Physical Layer (frequency

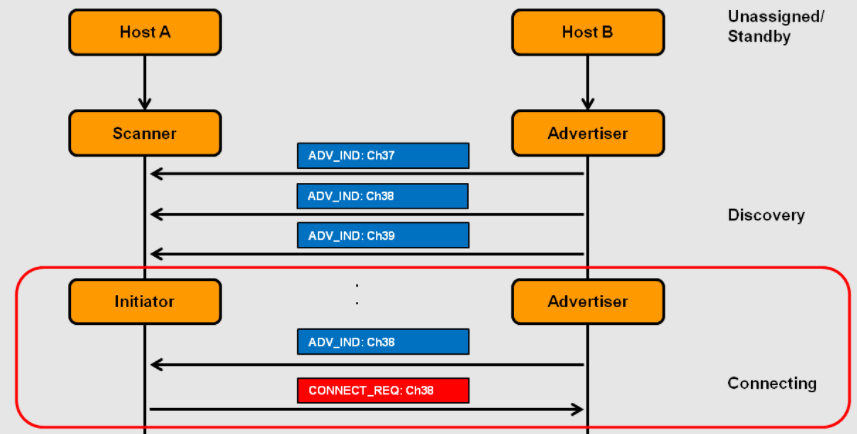
channel) will enable sending the frame to the remote device :



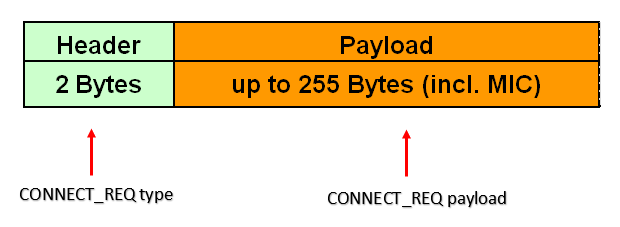
It is really important to notice that up to now **we only required Physical and Link Layer services** and that is all. That means only the Controller is needed to handle the advertising / scanning process. At this point, we are able to know the remote MAC device address so we can reach the remote device to initiate a connection process. We have to insist on that point : **getting the MAC address means having the opportunity to send the frame to the right device, but still there are still some parameters needed to be able to communicate. Indeed, BLE communications is based onTime Division Multiple Access (TDMA) which means both devices have to agree on communication time intervals, we are going to see it in the next part.**

**3.3. Connection Process**

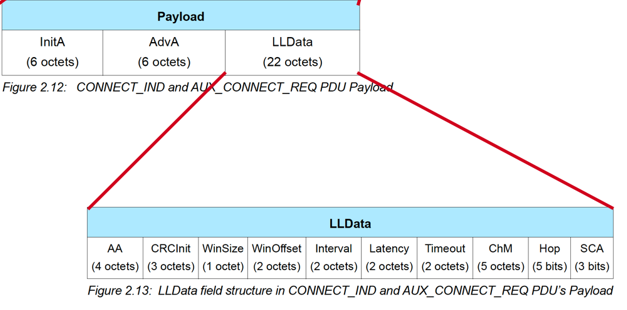
The connection process also has a major role to define how our two devices are going to use the medium. The connection process only requires an advertising channel packet too. Let’s remind the process and then study it deeply :



However in this case, the 2 Header bytes will code for a CONNECT\_REQ (standing for a connection request). :

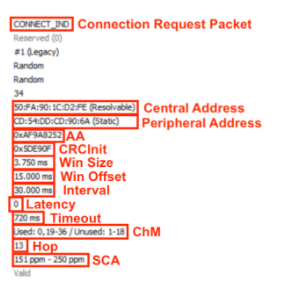


This connection request will use MAC addresses to reach out to the remote device. **More precisely, the connection request payload will set core communication parameters such as frequency hopping sequence, connection intervals and slave latency etc.These parameters make both Master and Slave agree on how they are going to communicate and what frame they should expect to.** We can see these parameters in picture below :



The Master is going to let the slave know about connection intervals which means “at which frequency they are going to send data” : this is where TDMA takes place.

Here again we can identify the role of the Link Layer that is “taking in charge the way both devices use the medium ”. On picture below, we can see an example of these parameters values :



However, the Data Link layer has other roles particularly in frame integrity checking.

We have to insist on a specific header : the MIC header. When Connection is set, data frames are exchanged. The Data Link layer is going to control frames integrity in terms of bits errors. This is done by using the MIC header. Thus, besides allowing frames exchanging, **the Data Link layer also participates in reducing frames errors** **and serv “clean” data to the upper layers.**

**3.4. L2CAP sublayer**

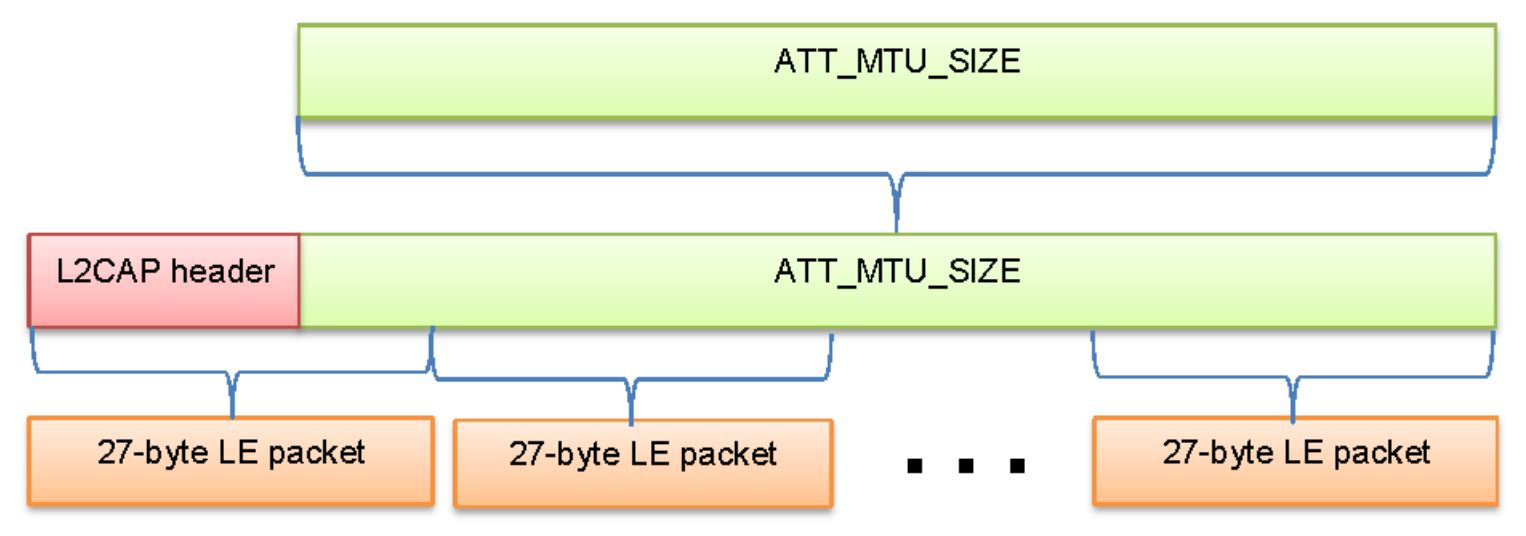
The L2CAP is a sublayer protocol that is in charge of passing the packets to the Host Controller Interface (HCl), that is to say between the controller and the host entities. It provides multiplexing between the higher layer protocols, segmentation and reassembly of packets, and QoS management for higher layer protocols.

In its basic mode, the L2CAP enables a packet configuration with a payload of 672 bytes as default, 48 bytes as minimum supported figure and 64k bytes as maximum figure.

L2CAP supports retransmission and flow control modes thanks to CRC checks, but also segmenting and reassembling packets when transmitted packets are too large for the limits of lower layers data packets. Reliability concerning all these modes is not fully guaranteed by this protocol, lower layers also bring additional guarantee.

In addition, there exists two L2CAP modes over those originally supported :

* Streaming Mode (SM) : no re-transmission or flow control
* Enhanced Retransmission Mode, ERTM : improved perfomances compared to original mode.

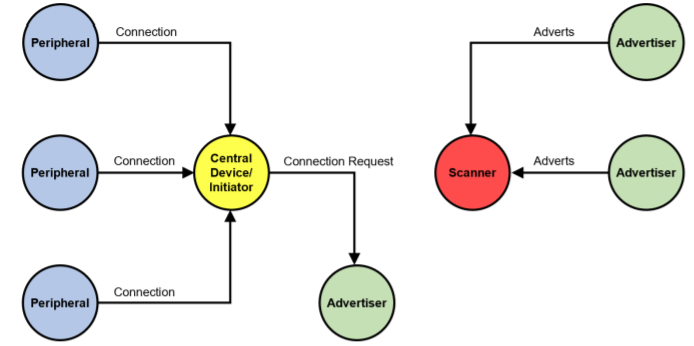


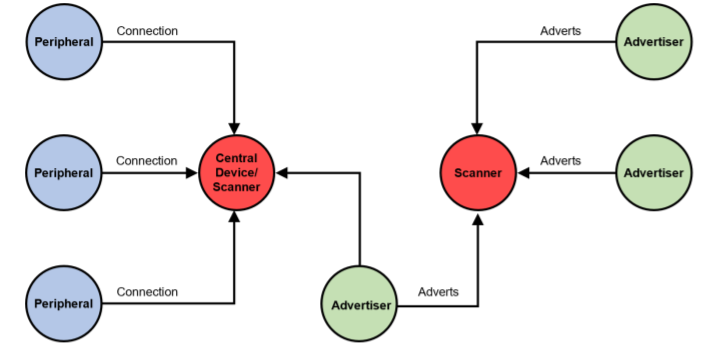
The figure above shows L2CAP packet fragmentation. As explained earlier, higher protocols (i.e. ATT) are allowed to use larger payload sizes, which obliges lower layers, such as L2CAP to practice fragmentation, and split a large packet into multiple smaller packets. We assume the size of L2CAP PDU is 27 bytes, the header is 4 bytes, we have a resulting default size for ATT\_MTU of 23.

**3.5. Network Topologies**

In BLE Technology, the devices can take one of the following roles :

* Advertiser: A device that broadcasts advertisement packets, but is not able to receive them. It can allow or disallow connections.
* Scanner: A device that only listens for advertisements. It can connect to an advertiser.
* Peripheral: A device connected to multiple central devices (BT 4.1 and newer).
* Central device: A device that is connected to one or more peripherals. Theoretically, a central device can have an unlimited number of peripheral devices connected to it, but in practice the central device can connect 4-20 peripherals at a time.
* Hybrid: It is possible for a device to advertise and scan at the same time or be connected to a central device and advertise or scan simultaneously. This is, however, vendor-specific, and the exact features that are supported should be checked with the vendor.

Examples of Bluetooth low energy topologies are shown in the following two figures.



The picture above shows a regular BLE network topology that demonstrates efficiently how important are the MAC layer et Physical Layer in communication exchanges.

**3.6. Limits**

Two main limits remain clearly important :

* distances aspect : BLE channel frequencies causes the communication distances to not exceed 50 m
* security aspect : during both the discovery and connection processes, no security mechanism was initiated by the devices. This means both MAC et Physical layers do not ensure secure connections. Security takes place on upper layers once the connection is set. Security mechanisms will be our subject for the next part.

**4. Security Mechanisms**

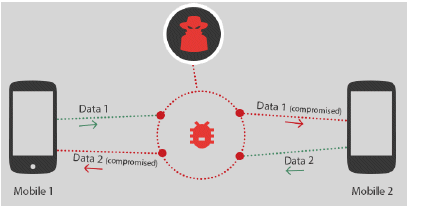
**4.1. Threats and vulnerabilities**

The purpose of the BLE protocol is to counter attack three major security issues.

1. **Passive eavesdropping** : Using a 2.4GHZ channel sniffer (say Wireshark for instance), it becomes possible to listen to the communication between BLE devices without the consent of communicating devices. If unencrypted messages or unsigned messages are used in communication, hackers can get direct access to all the confidential data exchanged between the devices. (Pairing procedures are the well-known techniques to avoid the eavesdropping issues and encrypt the messages before exchanging. However, if the attacker is listening to the devices during the pairing process itself, then pairing methods can’t assure the safety against the attack!).



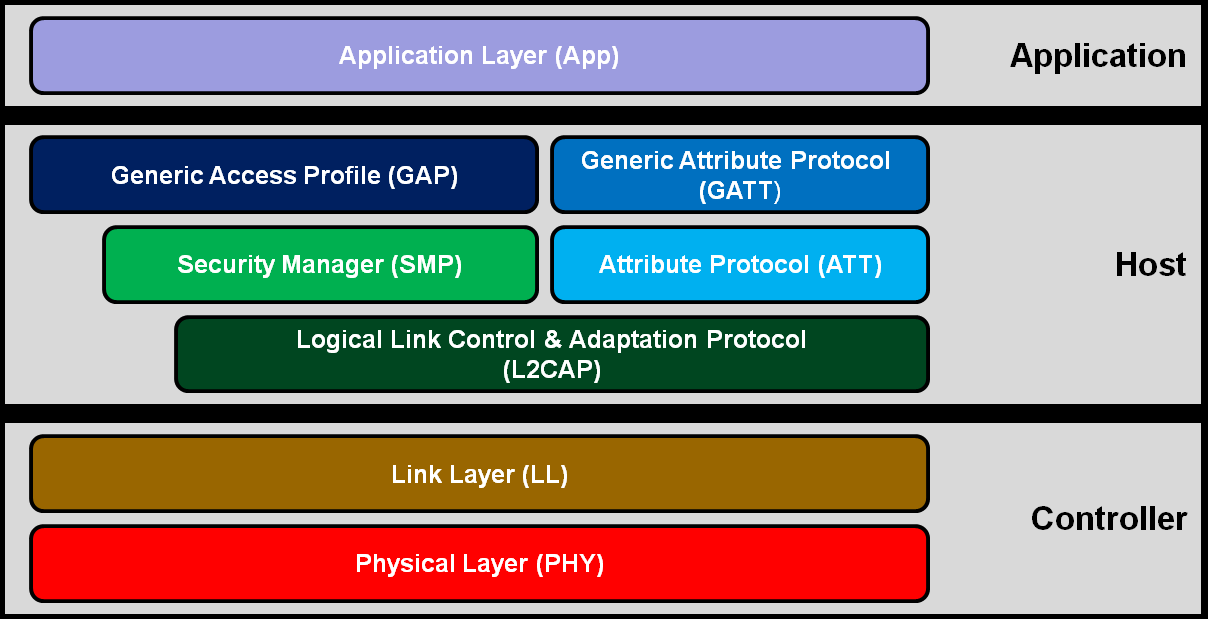
1. **Active eavesdropping** (also known as MITM issue standing for *“Man In the Middle”*) occurs when a third device interferes in the middle of the communication between the initiator and the responder. This third device receives emitted datas and can add fake informations inside the frame and re-transmit it to the responder (as depicted below) :



1. **Privacy and Identity tracking** : Bluetooth Low Energy 4.0 devices are designed for periodic advertisement of the status or its existence. Advertisement packets contain the MAC address of the broadcaster and unique service information. During this phase the datas are not encrypted within a key to protect the information and more specifically the address. That’s a major issue that makes BLE pretty vulnerable to cyber attacks.
2. **Duplication of devices replicating the MAC address** which is consequently due to the previous issue.

**4.2. Security options**

To deal with these security issues, BLE architecture deploys a third layer beyond the physical layer and the link layer which hosts several protocols that contain very private and confidential information about the device. Hence why it is mostly known as **the host layer.**



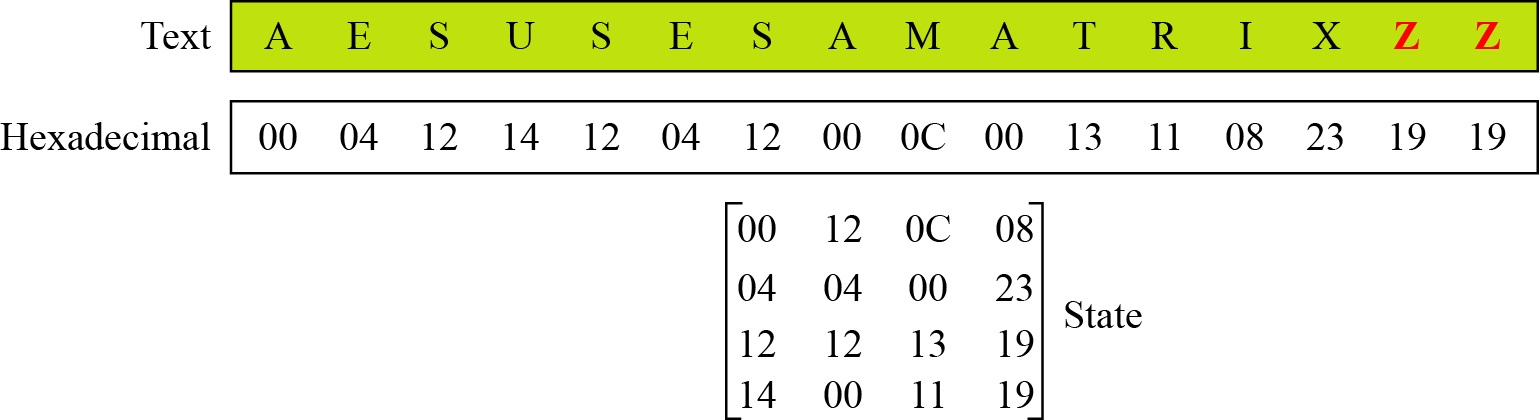
The host layer is the one layer that covers the security aspect. Let’s take an insight for a better understanding of how security is handled inside BLE architecture :

The Security Manager (SM) is the one layer that orchestrates and coordinates other protocols. The security manager defines the pairing (the first phase to establish connection), encryption and signing between the Bluetooth low energy devices. The GAP (for generic access profile) and GATT (for Generic Access Attribute) layers are in charge of the authentication used in the pairing phase and bonding phase, relying on the Attribute Protocol which will give access to identifiers, addresses and specificities of the device. This is responsible for all the security and privacy implementations of the BLE stack such as generation and storing various keys, generating random address and address resolution for the privacy feature. Security manager uses the services provided by the L2CAP layer to manage the security. Each device can generate its own key without any external influence and the strength of the key is proportional to the algorithm implemented in the device.

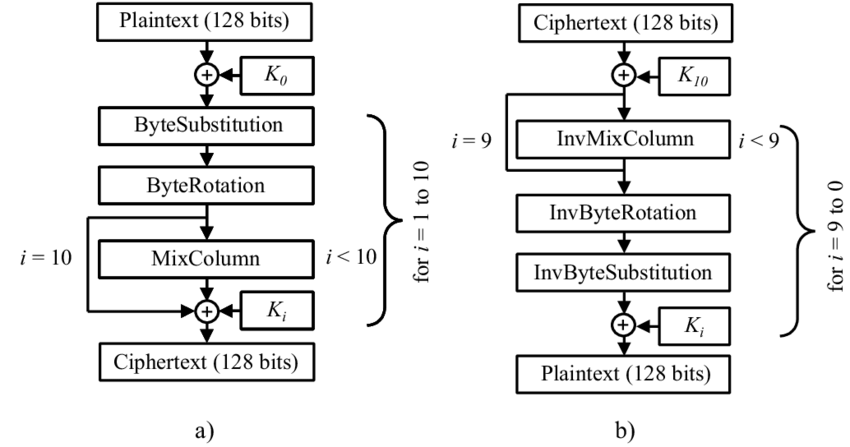
To overcome the cyberattacks previously tackled, security mechanisms are at stake.

The way that BLE overcomes the passive eavesdropping is by encrypting the data being transferred using **AES cryptography** (AES stands for Advanced Encryption Standard) also known as the *Rijndael cipher* cryptography.

We can briefly summarize this encryption method as following : To exchange and read datas, BLE devices need to define a password. This password is decided by generating a random sequence of 16 characters (ie. 16 bytes which makes in total 128 bits). This sequence is called the *plain text*. Each character of the plain text is then divided and distributed in a matrix composed of 16 sub-blocks, commonly designated as the *cypher blocks of* the *State matrix* (see picture below) :



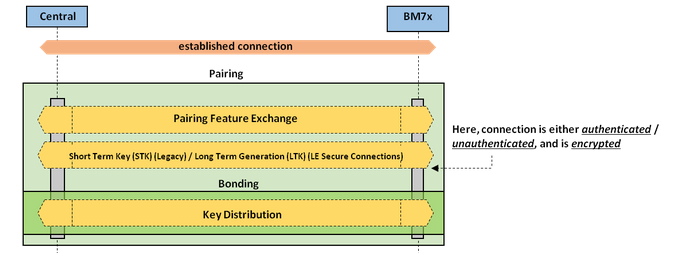
This state matrix will then follow a serial number of mathematical operations such as rotation, shifting, substitution, column mixture, inversion, multiplication, and overall XOR operation. These operations will be accomplished after multiple iterations (10 times generally) and more specifically, the XOR process will require ***Key matrices, called the*** sub-keys and noted Ki (i ranging from 1 to 10). At the very end, the algorithm will generate a *ciphertext (see picture a) )* which represents *the encrypted message over 128 bits.*



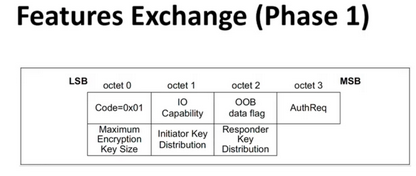
To decrypt the ciphertext, the receiver inverts the whole process used for encryption. However to do so and to obtain the original plaintext, the receiver needs all the subkeys Ki, usually assembled in one sole keystone that we usually note K *(see picture b) ).* K is the final and global key to decrypt the message throughout AES.

Note that today AES still remains one of the strongest mechanisms to encrypt datas and ensure security. Obviously, the strength of this algorithm strongly depends on different parameters such as the key length (128 bits in our example). Note that A PC that tries 255 keys per second needs 149.000 billion years to break AES.

=> The method by which the keys are exchanged in BLE communication refers to the **“pairing method”** or “association model”. In this setup, the devices exchange a Temporary Key (TK) and use it to create a Short Term Key (STK) which is used to exchange data on a temporary basis.



**Phase One** : This phase begins when the initiating device sends a ‘Pairing\_Request” to the other device. The two devices then exchange I/O capabilities (Input/Output), authentication requirements, maximum link key size and bonding requirements. Basically this phase consists of two devices exchanging their capabilities and determines how they will set up a secure connection. It is also important to note that all data being exchanged during this phase is **unencrypted** and that’s mostly when the cyberattacks might happen. Here is a scheme of the frame exchanged during phase 1.



Each device first determines its peer's input and output capabilities, selected from one of the following possibilities:

* No Input No Output
* Display Only
* Display Yes/No
* Keyboard Only
* Keyboard Display

These capabilities are communicated between the devices by using the Security Manager (SM) Pairing Request message. The Auth. Requirement flags can demand and include MITM protection.

**Phase Two** : Once phase one is complete, the devices generate and/or exchange the ***Temporary Key*** (TK) using one of the pairing methods. From there, the two devices then exchange Confirm and Rand values in order to verify that they both are using the same TK. Once this has been determined, they will use the TK along with the Rand values to create the STK. The STK is then used to encrypt the connection. It’s also important to note that the generation of the STK might be done within different methods, called the “pairing methods” (see next section).

**Phase Three:** After the pairing phase devices exchange a Long-Term-Key (LTK) to pre-configure the bounding after which the connection will be secure. This LTK will remain until the end of the communication to ensure encrypted datas exchange.

**Pairing Methods**

A pairing procedure involves an exchange of Security Manager Protocol packets to generate a temporary encryption key, the STK as depicted in the diagram above. During the packet exchange, the two peers negotiate one of the following STK generation methods.

### **Just Works**

The STK is generated on both sides, based on the packets exchanged in plain text. This method provides no security against Man-In-The-Middle (MITM) attacks.

### **Passkey Display**

One of the peers displays a randomly generated six-digit passkey and the other side as required to enter it. In certain cases both sides enter the key, if no display is available. This method provides protection against MITM attacks.

### **Out of Band (OOB)**

This method has the additional data transferred by means other than the BLE radio (such as another wireless technology like NFC). This method also provides protection against MITM attacks.

### **Numeric Comparison (Also Called LE Secure Connections Pairing in BLE v4.2)**

This method uses an algorithm called Elliptic curve Diffie–Hellman (ECDH) for key generation (another key-generation mechanism) and a new pairing procedure for the key exchange.

**5. Power consumption**

The main innovative characteristic of BLE, in comparison with Classic Bluetooth and other wireless technologies, is the low power consumption.

Indeed, both in broadcasting and connection mode, the RF module turns on to send or receive data and then turns off in order to save energy consumption.

Bluetooth affirms BLE has a current consumption, on average, lower thanZigBee.

STMicroelectronics provides a simulator to evaluate and measure the power consumption during all possible states of the BLE IC RF module.

To evaluate the power consumption of a BLE IC, it is necessary to evaluate the average current absorbed during the active phase in each modality of communication. In fact, on the other side, when the radio is in the sleep phase, the current consumption is approximately 1 µA (with 3 V of reference voltage level).

In broadcasting mode, the advertiser sends a packet on the three advertising channels in each advInterval.

Therefore, it is easy to notice that the consumption is directly proportional to the packet data length and to the number of channels used to communicate, while it is inversely proportional to the advertising Interval. In scanning mode, the average current consumption is only related to the Scan Window, which determines the active phase, and the scanInterval, which indicates the time between two active phases.

Moreover, the energy consumption during the active scanning is higher than the passive scanning because the advertiser is set to listen to other requests from the scanner. An analysis of the power consumption of a BLE broadcasting mode during the discovery process has been done by Liu et al. They measured in detail the current consumption in each phase during the advertising mode (using 3 V of reference voltage level). The results show that the current consumption in the active phase of advertising is ~10.9 mA. That means that the total current consumption during advertising, with the minimum advInterval of 20 ms, is ~2.3 mA.

A power analysis and a battery life estimation in advertising mode for a concrete application has been done by Fafoutis et al. They extrapolate the power profile and the energy consumption when transmitting a triple BLE advertisement, which goes from around 30 µJ to less than 40 µJ, depending on the transmission power (range from −21 dBm to 5 dBm). During the connection mode, it is more difficult to examine the power consumption because it is strongly variable, depending on several parameters, such as the packet payload, the connInterval, the number of slaves per master, the type of communication (one-way or round-trip), and so on.

The average current consumption, estimated for the minimum connInterval of 7.5 ms, is ~3200 µA, while the consumption with a connInterval of 2 s is about ~13 µA. The operating voltage for the BLE IC is 3 V.

**Conclusion on the energy consumption for BLE**

* Energy consumption of a BLE IC depends on multiple factors being the packet data length, the number of channels used to communicate, and the advertising Interval.
* In the sleep phase, BLE is very low energy consuming since it only consumes 3e-6 W for the minimum interval of 20ms which gives us an energy consumed of 3 nJ.
* In the advertising phase, BLE consumes on average 40 uJ for the active sending of 3 adv requests with a minimum advInterval of 7ms. The advertising packet size ranges from 8B to 39B long. So that means that the consumption ratio varies from 625 nJ/b to 128 nJ/b.
* In the connection (DATA) phase, BLE consumes 38uJ for a minimum CnxInterval of 7ms. The data packet size ranges from 10B to 265B long. So the consumption varies from 0.47 uJ/b to 18 uJ/b
* In average, BLE has a power consumption of 50mW when in transmission, and has a data rate of 1.4Mbps, so the average consumption ratio would be something around 35.7 nJ/b.

**Conclusion**

We have seen in this report all the features of the BLE technology, which is one of the most used technologies for IoT objects. It is important to notice that every layer has its importance and especially the DataLink layer, based on the MAC protocol, as it is the layer that will enable a good use of the medium to communicate with the remote device.

BLE protocol is a low power consumptive protocol, with some security issues but it remains one of the easiest protocols to understand and use.

***Bibliography***

1. Oller, Joaquim, Carla Gomez, et Joseph Paradells. « Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology », 26 June 2012.
2. Tosi, Jacopo, Fabrizio Taffoni, Marco Santacatterina, Roberto Sannino, et Domenico Formica. « Performance Evaluation of Bluetooth Low Energy: A Systematic Review ». *Sensors*, 13 december 2017.
3. Katsandres, Jim. « Bluetooth Low Energy – It Starts with Advertising ». Bluetooth.com. *Bluetooth Blog* (blog), 15 february 2017.<https://www.bluetooth.com/blog/bluetooth-low-energy-it-starts-with-advertising/>.
4. Horbenko, Yuliia. « How Secure Is the BLE Communication Standard? » Lembergsolutions.com, 4 march 2019.<https://lembergsolutions.com/blog/how-secure-ble-communication-standard>.
5. SILICON LABS. « UG103.14: Bluetooth® LE Fundamentals ». s. d.<https://www.silabs.com/documents/public/user-guides/ug103-14-fundamentals-ble.pdf>.
6. Oudji, Salma (PhD). « Analyse de la robustesse et des améliorations potentielles du protocole RadioFréquences Sub-GHz KNX utilisé pour l’IoT domotique ». Université de Limoges, December 2016.
7. Microchip. « Bluetooth® Low Energy ». Developer Help, 2021.<https://microchipdeveloper.com/wireless:ble-phy-layer>.
8. Afaneh, Mohammad. « A Deep Dive into BLE Packets and Events ». *BLE Technology, Tools* (blog), avril 2020.<https://www.novelbits.io/deep-dive-ble-packets-events/>.