BLUETOOTH LOW ENERGY E-RL 2 LOCK

PRODUCT SPECIFICATION

Draft Version *1.00*

1/03/2019

AXA Bike Security
EnergieStraat 2
3903 AV Veenendaal
Netherlands
http://www.axabikesecurity.com/
T: +31 318 536 111
F: +31 318 595 535

Disclaimer

The information contained in this document is believed to be correct and complete. However, AXA Stenman Nederland B.V. ("AXA") does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information or as to its suitability (for products) and shall have no liability whatsoever for the consequences of use of such information. In no event shall AXA be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

AXA's total liability for any direct damages incurred by the products described herein, if and when established by a competent court, shall be limited to the price of such products.

Customers are responsible for the design and operation of their applications and products using AXA products, and AXA accepts no liability whatsoever for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the AXA product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

AXA reserves the right to modify any of the documentation at any time and without notice. AXA assumes no responsibility for any infringements of patents or other rights of third parties that may result from the use of any product or documentation of AXA.

Copyright

FCC/SED Regulatory notices

Modification statement

AXA Stenman Nederland B.V. has not approved any changes or modifications to this device by the user. Any changes or modifications could void the user's authority to operate the equipment.
AXA Stenman Nederland B.V. n'approuve aucune modification apportée à l'appareil par l'utilisateur, quelle quén soit la nature. Tout changement ou modification peuvent annuler le droit d'utilisation de l'appareil par l'utilisateur.

Interference statement (if it is not placed in the device)

This device complies with Part 15 of the FCC Rules and Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.
Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes: (1) l'appareil ne doit pas produire de brouillage, et (2) l'utilisateur de l'appareil de droit accepter tout brouillage radioélectricque subi, méme si le brouillage est susceptibile d'en compromettre le fonctionnement.

Wireless notice

FCC
Radiation Exposure Statement: The device has been evaluated to meet FCC general RF exposure requirement. The device can be used in portable exposure condition with minimal separation distance (5 mm).

ISED
Radiation Exposure Statement: The device has been evaluated to meet ISED general RF exposure requirement. The device can be used in portable exposure condition with minimal separation distance (5 mm).

Déclaration d'exposition aux radiations: L'appareil a été évalué pour répondre aux exigences générales d'exposition aux radio fréquences. L'appareil peut être utilisé en condition d'exposition portable avec une distance de séparation minimale (5 mm).

FCC Class B digital device notice

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

CAN ICES-3 (B)/ NMB-3 (B)

This Class B digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de class B est conforme à la norme canadienne NMB-003.

Table of Contents

1. Introduction

The Bluetooth Low Energy Wireless E-RL 2 lock can be controlled directly through standard Bluetooth LE communication without the need for a proprietary communication stack. Bluetooth Low Energy sometimes referred to as Bluetooth LE or Bluetooth 5 is a new technology completely different and not compatible with Classic Bluetooth that has been around since 2001. Classic Bluetooth is well known for its use in hands free car kits and wireless earphones remote controls for GSM and smart phones.

The Bluetooth Low Energy (BLE) based E-RL 2 takes all complicated lock handling, timing and key management out of the hands of the application (app) builder and offers a simple straightforward standard KeySafe cloud API interface to control and monitor the E-RL 2. The E-RL 2 employs an extreme smart low power saving mode minimizing current consumption to maximize battery life while offering selective response to control commands. All this is done with only one sole purpose and that is to maximize battery life while minimizing the negative effects for the user.

1.1 Quick Overview of Bluetooth LE

Bluetooth Low Energy (BLE) is a Bluetooth technology which was released into the main Bluetooth standard in 2010 along with the Bluetooth Core Specification Version 4.0. Classic Bluetooth devices had suffered from being extremely energy greedy, and the need for a better energy management was necessary. BLE solved this problem and has since been used on most modern devices (all Apple iPhone since the 4S, Samsung phones since the Galaxy SIII, and many more), it's now part of every smartphone in the market. BLE allows us to use the bluetooth communication for small data exchanges and hence to avoid consuming as much energy. The latest Bluetooth standard has been released in June 2016 and since than being implemended in smartphones, the new standard is marketed as Bluetooh 5. The biggest improvements of the new standard are the extended frame length, extended range if implemented and the increased communication speed. The extended range and increased communication speed are mutually exclusive, meaning that you can't have both at the same time.

1.1.1 Application Perspective

Before we get into the description of how the E-RL 2 works in detail, let's simply review a couple characteristics of Bluetooth LE devices and how the E-RL 2 fits in these:

- **Servers vs. Clients**: these definitions are fairly classic. The Client emits requests and receives responses while the Server listens for requests and sends responses.

- **Central vs. Peripheral**: The peripheral device has valuable information to share with central devices. The central device treats the received data to fulfill specific tasks.

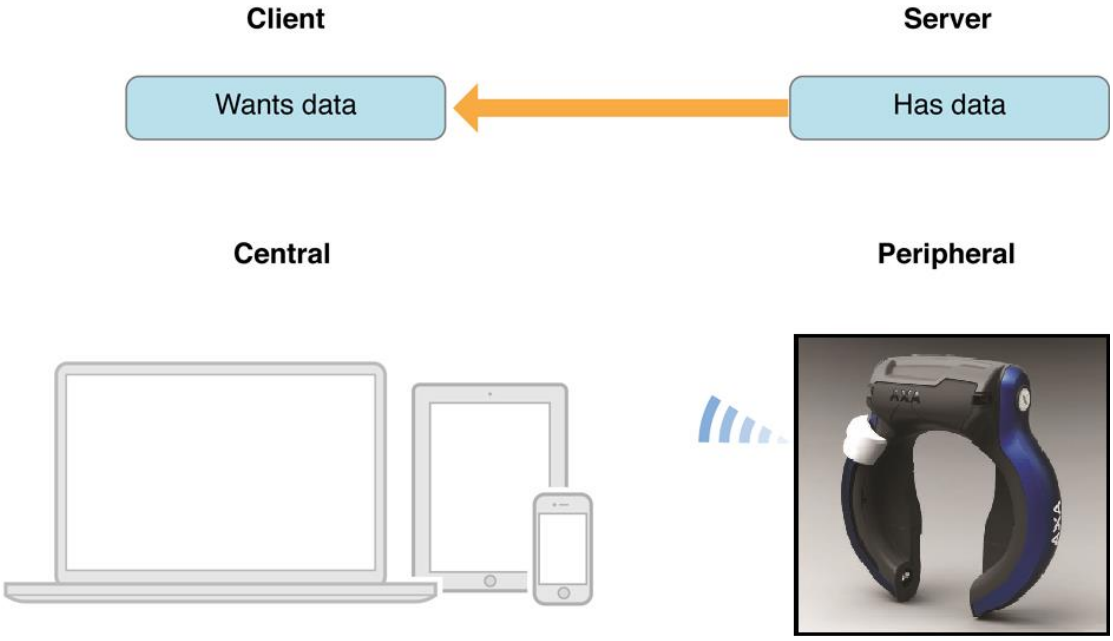

Figure 1.

Figure 1 is giving an overview of the different roles, so what is the difference between a **client** and a **server**? First let me remind you that a client and a server is not interchangeable with master/slave.

Figure 2.

Standard Bluetooth LE peripherals broadcast their presents in order to be found by centrals like smartphone's and tablets, see Figure 2. The E-RL 2 is no exception to this, when not connected by a central its broadcasting its presents so other centrals can find it during a scan and make a connection if the peripheral allows connections that is.

Apart from the higher power consumption this introduces a security risk for some product like bicycle locks, thieves will be able to locate bicycles in garages and/or sheds by simply using one of the many freely available Bluetooth LE scanner/localizer apps for smartphones. In order to overcome this kind of problems and to extend the battery lifespan the BLE E-RL 2 has the option to activate a smart sleeping mechanism, during a sleeping period it's not broadcasting and undetectable even for the bicycle owner's app.

Exiting this particular sleeping state the E-RL 2 needs to be woken-up by a simple button press on the side of the lock to start it advertising again during a set period. The smartphone app will now be able to detect the advertising packets and act accordingly depending on the relationship with the E-RL 2.

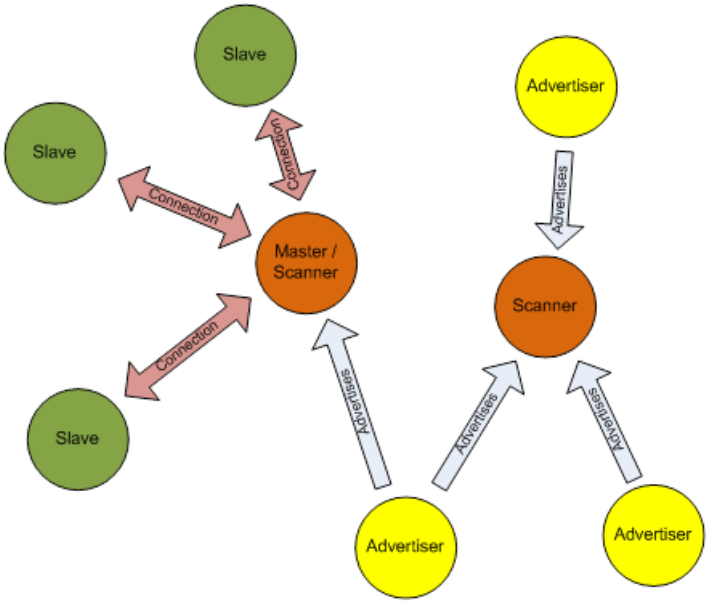

Figure 3.

In real-life situations smartphones (**Central**) will be surrounded by peripherals broadcasting (**Advertising**) their presents, see Figure 3.

A **master (or Central)** is the BLE device that initiates an outgoing connection request to an advertising peripheral device.

A **slave (or Peripheral)** is the BLE device which accepts an incoming connection request after advertising.

A slave can only be connected to one master, but a master can be connected to multiple slaves. In the smartwatch example, your iPhone can theoretically connect to multiple smartwatches at the same time. However, your smartwatch can only ever connect to one smartphone at a time.

There is no limit in the Bluetooth SIG on the number of slaves a master can connect to. Generally this will be limited by the BLE technology or Bluetooth stack you use.

Devices such as smartphones or tablets would generally (but not exclusively) adopt the role of "Scanner" and would "discover" other devices that have adopted the "Advertiser" role by a process called **discovery** which can be **active** ("are there any devices out there?") or **passive** ("I'll listen whilst devices advertise their presence"). Devices that adopt the "Advertiser" role are generally (but not exclusively) smaller footprint devices such as heart rate monitors or temperature sensors.

Once devices have discovered one another one will act as an "**Initiator**" (typically the smartphone or tablet type device) and attempt to connect to one of the devices that it has discovered. If successful it will adopt the role of "**Master**" and the other will adopt

the role of "**Slave**". "**Master**" devices will initiate commands and requests to "**Slave**" devices which will respond.

One of the first task of a Bluetooth LE application, such as on a smartphone app, is to discover other Bluetooth LE devices that it can connect to.

1.1.2 Services

Once a device has been discovered the next task is to figure out what services are offered by the device. So, what's a service? Well, a service consists of:

- A Service Specification, which consists of:
  - A collection of characteristics;
  - References to other service.

Figure 4 is giving an visualization to help cement the concept of services and characteristics. When talking about services we use the names:

- **GATT Server**
- **GATT Client**

This highlights the client/server model that is used at this level of the architecture.

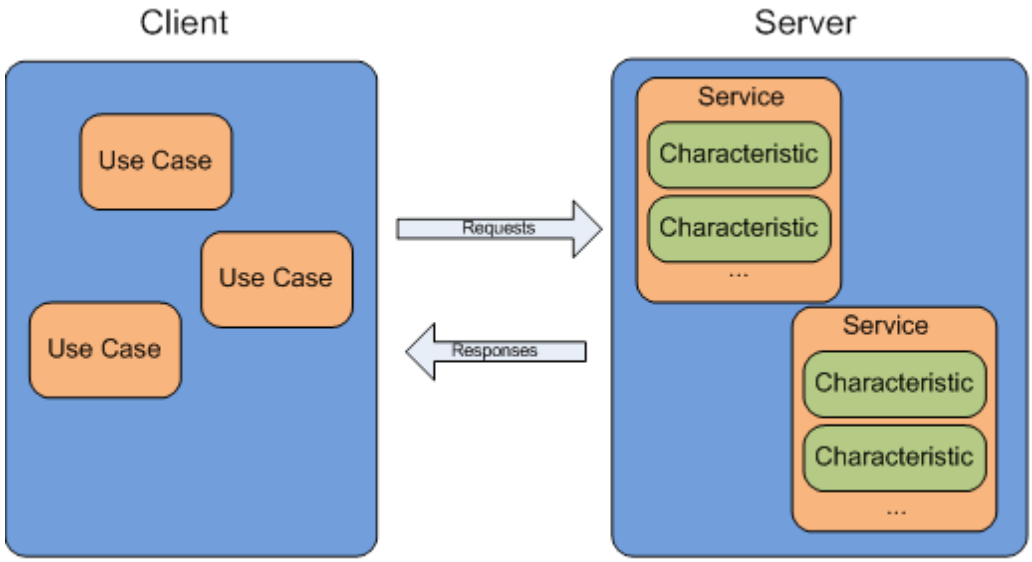

Figure 4.

Let's move on to the differences between a **GATT server** and a **GATT client**

A **GATT client** is a device which accesses data on the remote GATT server via read, write, notify, or indicate operations.

A **GATT server** is a device which stores data locally and provides data access methods to a remote GATT client.

You can easily see that it is possible for a device to be a GATT server and a GATT client at the same time. While it is most common for the slave (peripheral) device to be the GATT server only and the master (central) device to be the GATT client, this is not required. **The GATT functionality of a device is logically separate from the master/slave role.** The master/slave roles control how the BLE radio connection is managed, and the client/server roles are dictated by the storage and flow of data.

1.1.3 Profiles

A GATT database implements one or more **profiles**, and each profile is made up of one or more **services**, and each service is made up of one or more **characteristics**.

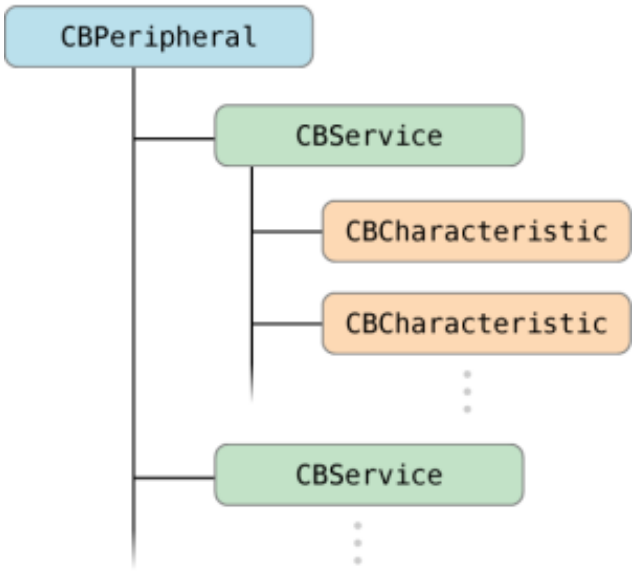

Figure 5.

GATT servers can implement as many profiles, services, and characteristics as needed by the product, figure 5. When using a non-standard profile, a 128bit UUID is required and must be provided by the peripheral producer offering this unique profile. You can also adhere to any **Bluetooth SIG profiles** (services, and characteristics) currently supported, in which case the profile UUID is 16bit long. Every BLE device acting as a GATT server must implement the official **Generic Access service**. This includes two mandatory characteristics: **Device Name** and **Appearance**.

1.1.4 Attributes and Characteristics

Remember that a service is made up of one or more **characteristics**. However, one characteristic may be comprised of many different **attributes**.

Each **attribute** is given a unique numerical **handle** which the GATT client may use to reference, access, and modify it. Every characteristic has one main attribute which allows access to the actual value stored in the database for that characteristic. When you “write a value to a characteristics” or “read the value of the characteristic” you are doing read and write operations on the main data attribute (of said characteristic).

Some other related attributes are read-only (such as a **Characteristic User Description** attribute), some control the behavior of the characteristic (such as the **Client Characteristic Configuration** attribute which is used to enable **notify** or **indicate** operations).

Every attribute has a UUID. These may be either 16 bits (e.g. **"0x180A"**) or 128 bits (e.g. **"00001530-1212-EFDE-1523-785FEABCD123 "**). All 16-bit UUIDs are defined by the Bluetooth SIG and are known as adopted UUIDs. All 128-bit UUIDs are custom and may be used for any purpose without approval from the Bluetooth SIG. Two very common 16-bit UUIDs that you will see are **0x2901**, the Characteristic User Description attribute and **0x2902**, the Client Characteristic Configuration attribute (to enable either **"notify"** or **"indicate"** on a characteristic).

One important note is that some attribute UUIDs do not technically need to be unique. Their **handles** are always unique, but the UUIDs occasionally overlap. For example, every Client Characteristic Configuration attribute has a UUID of **0x2902**, even though there may be a dozen of them in a single GATT database.

1.1.5 Advertising and Connections

The Generic Access Profile (**GAP**) specifically describes behaviors and procedures for device discovery, connection establishment, security, authentication, and service discovery, and this along with performing a single defining role. In essence, a Bluetooth device may incorporate either initiating or accepting procedures, and the peer device must support the corresponding functionality. One of the most important things to understand with Bluetooth low energy is how two devices first find one another, work out what they can do with one another, and how they can find and connect with one another repeatedly. This is really what GAP defines.

There are four GAP roles defined for a Bluetooth low energy device:

- **Broadcaster**
- **Observer**
- **Peripheral**
- **Central**

A broadcaster is a device that sends advertising packets. Typically, this is used to broadcast some data from a service to other devices that happen to be in an observer role. A broadcast must have a transmitter but does not need a receiver. A broadcast-only device, therefore, only needs a transmitter.

An observer is a device that scans for broadcasters and reports this information to an application. An observer must have a receiver; it can also optionally have a transmitter.

A peripheral is a device that advertises by using connectable advertising packets. As such, it becomes a slave once connected. A peripheral needs to have both a transmitter and a receiver. The E-RL 2 has adopted the role of peripheral device and as such is sending advertising packets while not connected.

A central is a device that initiates connections to peripherals. As such, it becomes a master when connected. Just like a peripheral, a central needs to have both a transmitter and a receiver. We explained before that since the E-RL 2 has adopted the role of peripheral device the smartphone will need to accept the role of central in this system. The role that is already normal for smartphone's since most other BLE peripherals connected to the smartphone are peripheral's, for example smartwatches or other body sensors that measure vital signs while cycling. A device can support multiple GAP roles at the same time. For example, a device can be a broadcaster and a peripheral at the same time.

1.1.6 Pairing and Bonding

Just a quick write up on the difference between pairing and bonding, since these terms get used interchangeably. This has to do with the usage of 'pairing' in Bluetooth Classic, or BR/EDR.

As far as Bluetooth LE is concerned, pairing and bonding are two very distinct things. The short explanations are that pairing is the exchange of security features each device has, and creating temporary encryption for the live cycle of the connection. Bonding is the exchange of long term keys **after pairing has occurred**, and **storing those keys for later use**. Pairing is not the creation of permanent security between devices, which is called bonding. Pairing is the mechanism that allows bonding to occur.

Pairing

Pairing is the exchange of security features. This includes things like i/o capabilities, requirement for man-in-the-middle protection, etc. The client side begins this exchange. The client essentially says 'hey, i'd like it if you had these features'. The server replies, 'yeah, well, this is what I can do'. Once this exchange is made, the security that will be used has been determed. For example, if a server supports just noInput/noOutput for i/o capabilities, the Just Works pairing mechanism is going to be used. Once the pairing feature exchange is complete, a temporary security key is exchanged and the connection is encrypted, but only using the temporary key. In this encrypted connection, long term keys are exchanged. These keys are things like the (long term) encryption key to encrypt a connection, and also things like a digital signature key. The exact keys exchanged are determined by the security features of each device.

Bonding

This really just means that after the pairing features exchange and the connection has been encrypted (these two together are called 'pairing'), and keys have been exchanged, the devices store and use those keys the next time they connect. Keys can

be exchanged using the bonding procedure, but that does not mean they are bonded if the keys are not stored and used the next time.

If a device is bonded with another device, like a heart rate monitor and a smartphone, they can encrypt the connection without exchanging any sensitive security information. When the smartphone connects to the heart rate monitor, it can just issue a 'turn on encryption' request, and both sides will use the keys already stored, so nobody snooping can see a key exchange and therefore decode the messages being sent, as is done when pairing.

Certificate

Standard BLE does not use certificates for setting up a secure connection between the master and slave, the KeySafe enabled E-RL 2 does however use certificates signed by the cloud certificate authority KeySafe for setting up secure connections or creating secure relationships. Without the certificate handover by the master (e.g. smartphone) and positive outcome of the verification the E-RL 2 will not allow any device to pair, all pairing requestes by the master will be rejected with an insufficient authentication error code. When the certificate is accepted by the E-RL 2 it will initiate the setup of a secure link between the devices and allows the user to input the 6 digit passkey on older smartphones or on the more modern smartphones allows the app to input the 128 bit passkey for the user automatically. Entering the passkey is only required once during pairing and bonding, all other times the smartphone will remember the saved bonding information and connects flawlessly with the E-RL 2 until the certificates time has expired. Both the certificate and passkey are provided by the KeySafe cloud service upon requesting for an eKey. Next to the standard pairing and bonding the use of the KeySafe cloud API also does offer a secure relationship based on OTP (one-time-pass) commands. Using OTP's does not require the mobile device to pair and bond to securely command the lock to operate. The advantage is trivial, no need to enter passkeys or solving the many problems related to pairing and bonding in many iOS and Android smartphones due to software bugs in the smartphones BLE stack or software version specific particularities. The exact process of requesting, processing and handling eKeys and OTP's is explained in more detail in the KeySafe-Cloud end node API documentation.

1.1.7 Radio Communication

Classic and Bluetooth LE operates in the 2.45 GHz band which it shares with Wi-Fi, Zigbee and microwave ovens worldwide!. Bluetooth LE still retains its fundamental resilience by splitting its radio traffic across 40 channels as shown below (Figure 6).

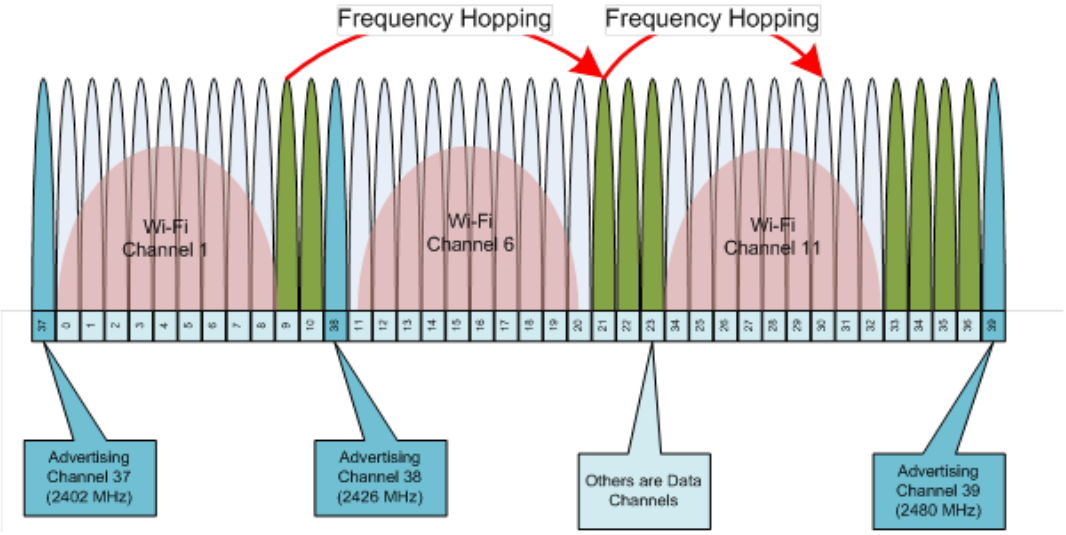

Figure 6.

The Bluetooth low energy channels differ from classic channels because of the relaxed modulation index. This means that the radio energy for each channel is spread wider; therefore, to prevent interference between adjacent Bluetooth low-energy channels, they are separated by 2MHz, instead of the classic 1MHz. In the Link Layer, these channels are divided into two types: advertising channels and data channels. These channel types are aligned with the advertising packets and data packets, as described earlier. When a packet is transmitted, if the packet is sent on an advertising channel, it is an advertising packet. If the packet is sent on a data channel, it is a data packet. There are 3 advertising channels and 37 data channels, as shown in Figure 6 (the advertising channels are rendered in light blue shading). The 3 advertising channels are not all placed in the same part of the ISM band because that would mean that any deep fade in a single part of the band would stop all advertising. Instead, the advertising channels are placed a minimum of 24MHz apart from one another.

The advertising channels are placed strategically away from significant interferers such as a Wi-Fi access point. These public access points typically use one of three 802.11 channels, either channel 1, channel 6, or channel 11. These channels have center frequencies of 2412MHz, 2437MHz, and 2462MHz and a width of approximately 20MHz. This means that channel 1 extends from 2402MHz to 2422MHz, channel 6 extends from 2427MHz to 2447MHz, and channel 11 extends from 2452MHz to 2472MHz. The advertising channels are placed at 2402MHz, 2426MHz, and 2480MHz. This means that the first advertising channel is below Wi-Fi channel 1, the second advertising channel is between Wi-Fi channel 1 and channel 6, and the third advertising channel is above Wi-Fi channel 11. This is illustrated in Figure 6, in which 3 Wi-Fi channels have blocked the use of data channels 0 to 8, 11 to 20, and 34 to 32. The 3 advertising channels, 37, 38, and 39, are all interference free. Bluetooth LE Radio traffic hops around these channels in a pseudo random manner so that the data with get through even though it's in an areas shared by a number of Wi-Fi networks, or microwave ovens. One of the differences between Bluetooth LE and classic Bluetooth is the number and use of these channels.

When in a data connection, an adaptive frequency-hopping algorithm is used. Adaptive frequency hopping makes it possible for a given packet to be remapped from a known bad channel to a known good channel so that the interference from other devices is reduced. To do this, a channel map of good and bad channels is kept in both devices. If the channel that would have been chosen by the master device is a good channel, then that channel is used; if the channel that would have been chosen is a bad channel, then it is remapped onto the set of good channels. A minimum of two data channels must be marked as good by a master.

Suppose, for example, that a Bluetooth low energy device is in the same area as a Wi-Fi channel 1 access point that is streaming data to another Wi-Fi device. The Bluetooth low energy device would mark Link Layer data channels 0 to 8 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,

1.2 System Description

Making the complicated system setup of the trusted third party KeySafe cloud service and its operation clear a visualization in Figure 7 is helping cement the concept of the KeySafe cloud service. The renting app on the smartphone is detecting the presents of bikes available for renting and offering this to the customer. The customer is making his or her selection and the app is sending renting details to the renting companies' backend computer system for availability. When approved by the renting agent backend computer system an specific eKey type will be requested for this particular bike from the KeySafe, the cloud computer is generating a certificate holding information regarding the bike, renting time and type of eKey. The KeySafe cloud will forward this information including a freshly generated passkey in case of pairing and bonding eKey type to the requesting rental agent backend computer who will forward this information to the renting app.

Figure 7.

It's up to the renting app to present the certificate to the E-RL 2 BLE service on the selected bike for verification and approval and send the passkey using an API call to the OS on the smartphone. When the certificate is accepted by the E-RL 2 and depending on the type of eKey a secure pairing/bonding sequence will follow providing

a permanent bond during the specified renting time period. When the renting period has expired it will not automatically close the E-RL 2 but will not allow the renting customer to unlock the Bike anymore. The permanent bond with the smartphone is not transferable to a different smartphone since its using unique session keys exchanged during the pairing and bonding process.

In case an OTP eKey certificate is requested from the KeySafe cloud the eKey is returned including a list of OTP commands used to lock and unlock, each of these OTP commands can only be used once and are transferable and not bound to the smartphone that initially send the eKey certificate. OTP eKey certificate are mainly but not exclusively intended for smartphones that do not allow the app to enter the passkey through an API call to the OS. In the KeySafe-Cloud end node API documentation a more detailed and thorough explanation of the different eKey types and there working is given.

1.3 New functionality in eRL 2

The functionality of E-RL 2 has been increased by smart plugin detection, always close push button functionality making it possible to close the E-RL 2 in all cases including the case the mobile phone's battery is dead. Figure 8 is showing the KeySafe API extensions between the E-RL 1 and E-RL 2.

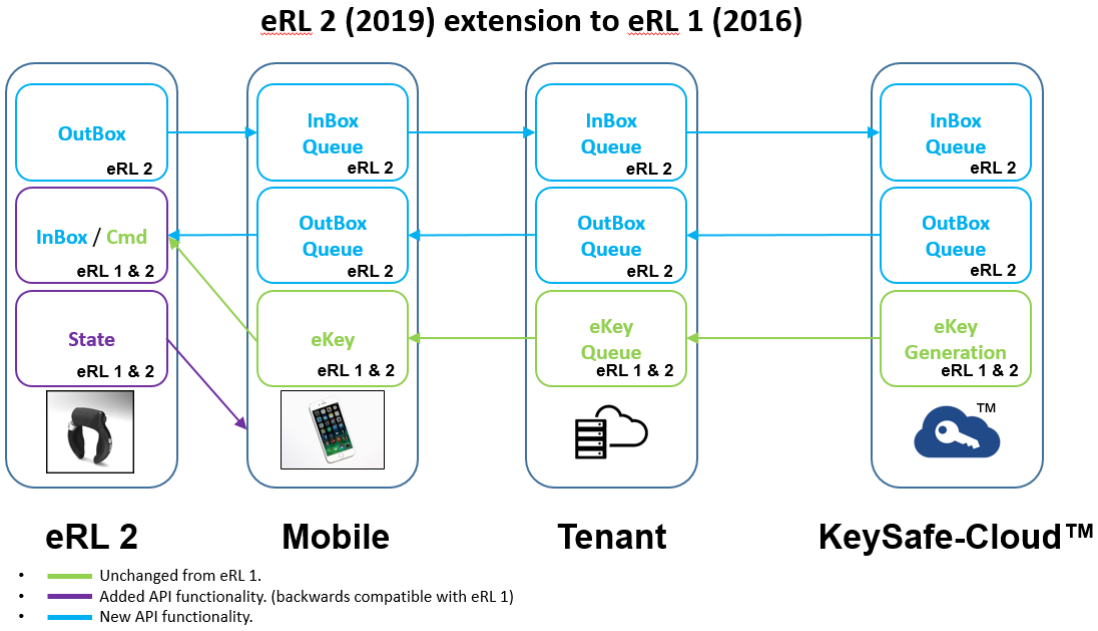

Figure 8.

The E-RL 2 makes use of the latest KeySafe-Cloud end node API interface giving full control over the fleet from your own backend software. It's possible to update the E-RL 2 from within your mobile App during use without the customer being aware, no interruptions in operation while updating.

2. E-RL 2 Profiles

The E-RL profile is offering the following 5 Standardised and non-standard services:

- The Standardised Service, which consists of:
  - **Generic Access Service**. (Mandatory for all BLE devices)
  - **Generic Attribute Service**. (Mandatory for all BLE devices)
  - **Device Information Service**.

- The Non-standard Service, which consists of:
  - **KeySafe Cloud Service**.
  - **Device Firmware Update Service**.

All services are explained in the following sections and furthermore specified in detail in the Appendices of this document.

2.1 Device Information Service

The Device Information Service is a standardized Bluetooth service that exposes the E-RL 2 device specific information characteristics including the following version specific information:

- Manufacturer Name String:
  - “Axa Stenman Nederland BV”
- Model Number String:
  - “E-RL 2”
- Serial Number String:
  - “2EA0A-5780B-AD92E-3FEDC”
- Hardware Revision String:
  - “V1.54”
- Software Revision String:
  - “V1.00”

This information shall be used by the client application (e.g. app) on how to proceed with the connection and which services and features can be expected from the E-RL. The serial number string contains the number which relates to the KeySafe cloud server, the number does have a different format compared to the same number broadcasted by advertisements to make it more and better readable for humans. The hardware revision string is relating to the BLE PCB inside the E-RL 2 only and is used

together with the software and firmware revision strings during the device firmware update process.

2.2 KeySafe Cloud Service

The device specific E-RL 2 KeySafe Cloud Service is a non-standard Bluetooth service that exposes an interface for the operation and control of the E-RL 2. In the KeySafe-Cloud end node API documentation a more detailed and thorough explanation of the operation of this vital service is given.

2.3 Device Firmware Update Service

The device specific E-RL 2 Device Firmware Update Service is a non-standard Bluetooth service that exposes a secure interface for FOTA. Currently a special app is required to update the internal software, the app will upload the new software and/or firmware after which the E-RL 2 will be update and ready again for use. This updating process takes less than 2 minute and is secured through the KeySafe-cloud.

3. Electrical Characteristics

Absolute Maximum Electrical Ratings(†)

Characteristic	
Ambient temperature under bias	-25°C to +85°C
Ambient temperature during operation	-20°C to +55°C
Supply Voltage Vdd pin with respect to Vss (Idle)	3.7V
Supply Voltage Vdd pin with respect to Vss (Running)	3.7V
Supply Voltage Vdd pin with respect to Vss (Stall)	3.7V
Supply Current Vdd pin	250mA
Peak (10s) Supply Current Vdd pin @ 3V (Stall)	500mA
ESD protection on all pins (HBM; MM)	≥ 2kV; ≥ 400V

Standard Electrical Operating Conditions (unless otherwise stated)

Operating temperature -20°C ≤ TA ≤ +55°C

Characteristic	Min	Typ.	Max	units	Conditions
Supply Voltage (Vdd)	2.2	3	3.6	V	Normal Mode
Supply Voltage (Vdd)	2.5	3	3.6	V	BLE FOTA Mode
Supply Current		90		mA	Motor Running
Peak Supply Current			180	mA	Motor Startup < 100ms
Motor Stall Current			500	mA	Motor Stall < 10s @ 3V
Power-saving Current[1]		19	25	µA	Vdd = 3V, (Sleep mode)
Start-up Time		290	500	ms	Power up
Unlocking Time	1.5	1.8	2.1	s	Locked -> Unlocked
Locking Timeout	14.9	15	15.1	s	Unlocked -> Unlocked
Stall Timeout	9.9	10	10.1	s	Vdd = 3V
Open/Closings		25000			On one CR123A Battery
Battery operation life		2		years	10 Open/Closings a day