

Bluetooth Pairing Part 1 – Pairing Feature Exchange

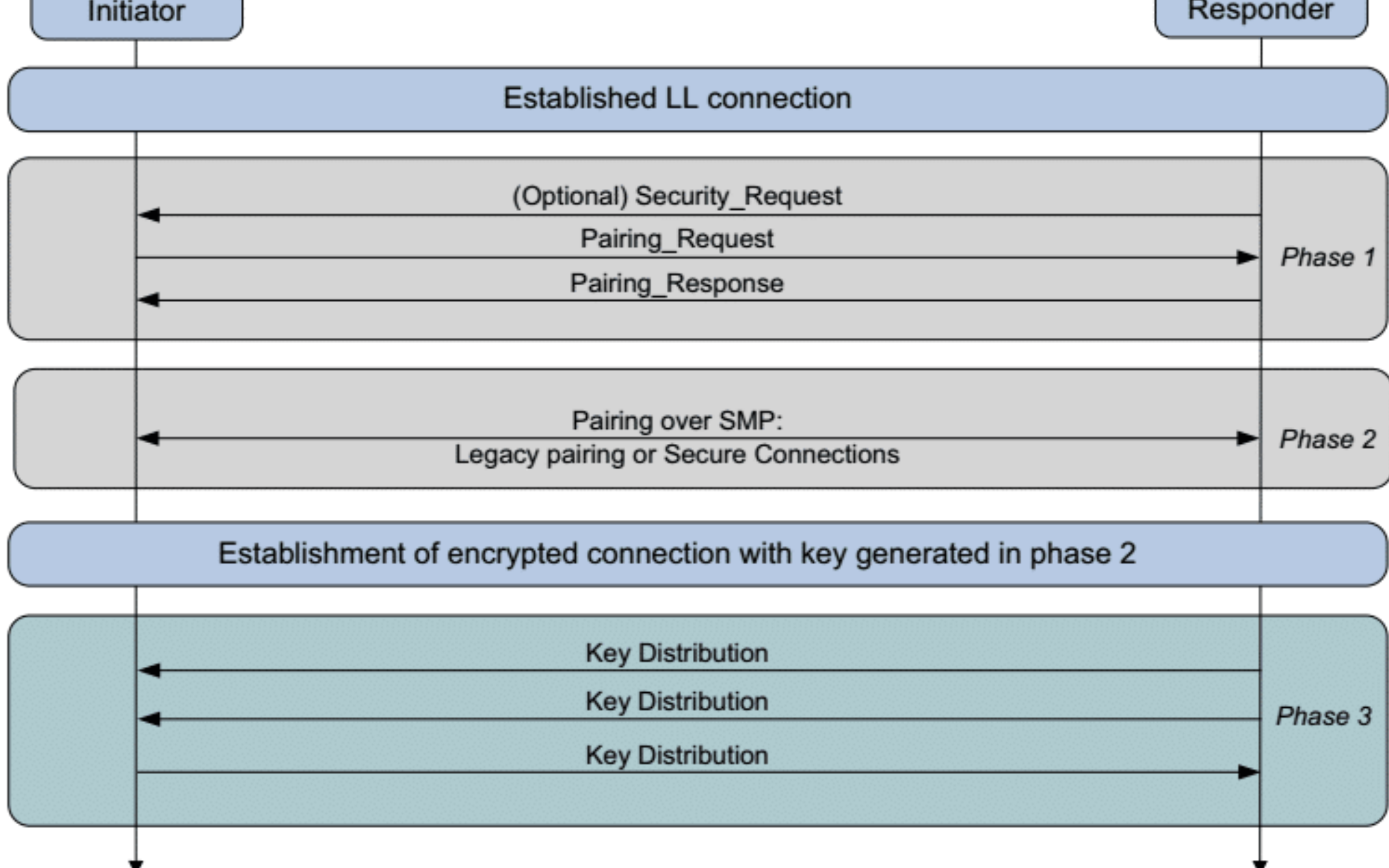
📅 March 29, 2016 | 👤 Kai Ren | 📁 Bluetooth Low Energy

In the *Bluetooth®* Core specification, there are three major architectural layers: Controller, Host and Application. In the Host Layer, there is a module called Security Manager (SM) which defines the methods and protocols for pairing and key distribution, the corresponding security toolbox, and the Security Manager Protocol (SMP) which defines the pairing command frame format, frame structure and timeout restriction. The Security Manager (SM) uses a key distribution approach to perform identity and encryption functionalities in radio communication.

Pairing is performed to establish keys which can then be used to encrypt a link. A transport specific key distribution is then performed to share the keys. The keys can be used to encrypt a link in future reconnections, verify signed data, or perform random address resolution. In general, there are 3-phase for pairing.

- Phase 1: Pairing Feature Exchange
- Phase 2 (LE legacy pairing): Short Term Key (STK) Generation
- Phase 2 (LE Secure Connections): Long Term Key (LTK) Generation
- Phase 3: Transport Specific Key Distribution

LE legacy pairing and LE Secure Connections may be new terms to most. LE is short for “low energy” and is in the Bluetooth® specification as a main feature of Bluetooth 4.0 and above. In the Bluetooth 4.2 specification, the Secure Connections feature to the LE physical transport was added, which upgraded pairing to utilize FIPS-approved algorithms (AES-CMAC and P-256 elliptic curve) on the Bluetooth LE physical transport. In order to distinguish Secure Connections from LE pairing as defined in the Bluetooth 4.0 and 4.1 spec, it is referred to as LE legacy pairing. Figure 1 is a pairing flowchart which applies to both legacy pairing and secure connections.



Today, we will look at Phase 1: Pairing Feature Exchange. Pairing is the exchange of security features that include things like Input/Output (IO) capabilities, requirements for Man-In-The-Middle protection, etc. The exchange of pairing information between two devices is done through the Pairing Request and Pairing Response packet. The contents of these two messages is shown below in Table 1 Pairing Request/Response.

Field	Initiator	Responder
Code	0x00	0x00
IO Capabilities	0x00	0x00
OOB DF	0x00	0x00
Bonding Flags	0x00	0x00
MITM	0x00	0x00
SC	0x00	0x00
Key Distribution	0x00	0x00

“Code”IO Cap, “IO Capabilities”

Code	0x00
------	------

Since IO refers to Input/Output (IO) capabilities, requirements for Man-In-The-Middle protection, etc. The exchange of pairing information between two devices is done through the Pairing Request and Pairing Response packet. The contents of these two messages is shown below in Table 1 Pairing Request/Response.

Field	Initiator	Responder
Code	0x00	0x00
IO Capabilities	0x00	0x00
OOB DF	0x00	0x00
Bonding Flags	0x00	0x00
MITM	0x00	0x00
SC	0x00	0x00
Key Distribution	0x00	0x00

For Output Capabilities, it could be “No Output” or “Numeric Output,” detailed below.

Field	Initiator	Responder
Code	0x00	0x00
IO Capabilities	0x00	0x00
OOB DF	0x00	0x00
Bonding Flags	0x00	0x00
MITM	0x00	0x00
SC	0x00	0x00
Key Distribution	0x00	0x00

After combined those capabilities of Input and Output, here is a matrix defining what IO capabilities the Bluetooth device should have.

IO Capabilities	Input	Output
Yes/No	0x00	0x00
No Input	0x00	0x00
Yes/No	0x00	0x00
No Output	0x00	0x00
Yes/No	0x00	0x00
No Input	0x00	0x00
Yes/No	0x00	0x00
No Output	0x00	0x00

None of the pairing algorithms can use Yes/No input and no output, therefore, “NoInputNoOutput” is used as the resulting IO capability.

From the above matrix, you map the corresponding IO capabilities and select below enum to place into Pairing Request/Response packet.

IO Capabilities	Input	Output
Yes/No	0x00	0x00
No Input	0x00	0x00
Yes/No	0x00	0x00
No Output	0x00	0x00
Yes/No	0x00	0x00
No Input	0x00	0x00
Yes/No	0x00	0x00
No Output	0x00	0x00

OOB DF, “OOB Data Flag”

OOB, or Out-of-Band, uses an external means of communication to exchange some information used in the pairing process. The OOB media could be any other wireless communication standard which can carry the corresponding information for pairing, like NFC or QRCode.

OOB DF	0x00
--------	------

BF, “Bonding_Flags”

Bonding is the exchange of long-term keys after pairing occurs, and storing those keys for later use — it is the creation of permanent security between devices. Pairing is the mechanism that allows bonding to occur.

Bonding Flags	0x00
---------------	------

“MITM”

MITM is short for “Man-In-The-Middle.” This field is a 1-bit flag that is set to one if the device is requesting MITM protection. This blog focuses on the procedure for the pairing feature exchange—if you are interested in MITM, please refer to the Bluetooth Core Specification v4.2, Vol1, Part A, 5.2.3.

“SC”

The SC field is a 1-bit flag that is set to one to request LE Secure Connection pairing. The possible resulting pairing mechanisms are if both devices support LE Secure Connections, use LE Secure Connections and otherwise use LE legacy pairing. So this flag is an indicator to determine Phase 2 pairing method.

“KC”

The keypress field is a 1-bit flag that is used only in the Passkey Entry protocol and is ignored in other protocols. Passkey Entry protocol is a typical pairing method of Legacy Pairing and Secure Connection. We will go into this in the next blog article.

“Maximum Encryption Key Size”

The maximum key size shall be in the range 7 to 16 octets.

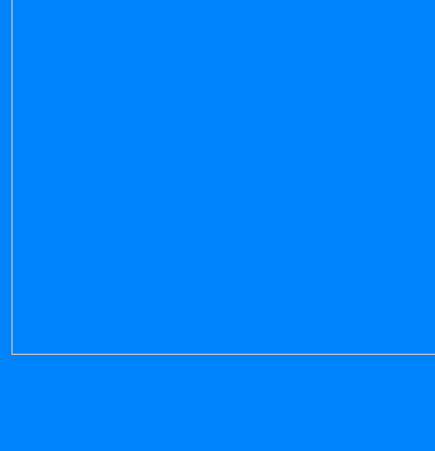
“Initiator Key Distribution” & “Responder Key Distribution”

These two fields have the same definition as below. I will explain when we talk about key distribution in the future series blog.

Initiator Key Distribution	0x00
Responder Key Distribution	0x00

When the exchange of pairing feature starts, the initiator and responder will exchange their pairing feature information with each other through pairing request and response. With the information, the initiator and responder can determine the I/O capabilities with each other, which pairing mechanism—legacy pairing or secure connection—should be used, and select the pairing method—**Just Work, Passkey Entry, Numeric Comparison** or **Out of Band**—to use in Phase2. We will explore the details in [Part 2: Pairing Method and Key Generation](#).

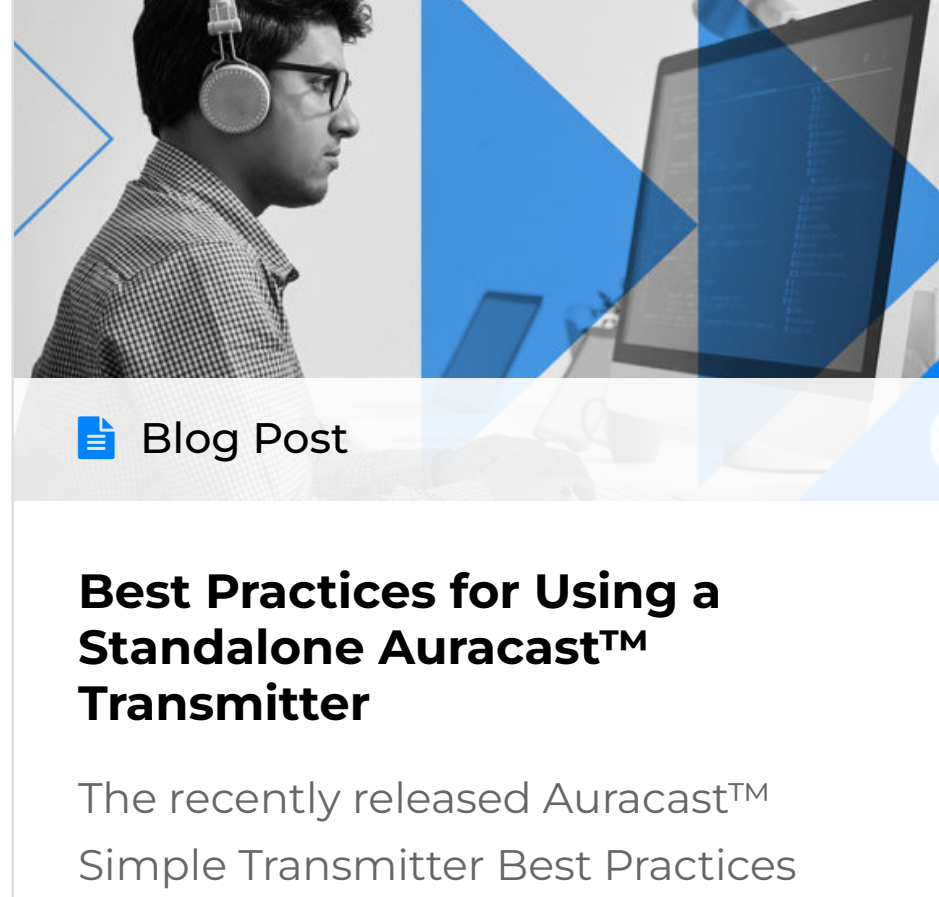
FEATURED DOWNLOAD



An Introduction to Bluetooth Low Energy Development

This self-study educational resource covers both theory and practice of Bluetooth Low Energy GAP and GATT application development. The guide will equip you with a solid understanding of key Bluetooth Low Energy concepts before guiding you through a series of software development projects that will allow you to put the theory into practice.

INSTANT DOWNLOAD 📄



📄 Blog Post

Best Practices for Using a Standalone Auracast™ Transmitter

The recently released Auracast™ Simple Transmitter Best Practices Guide describes a typical, qualified implementation...

[READ MORE](#) 🔗



📄 Blog Post

Auracast™ Broadcast Audio Introduces New Opportunities for Product Developers & Public Locations

Bluetooth® technology recently introduced a new Bluetooth capability, Auracast™ broadcast audio, that will deliver life-changing...

[READ MORE](#) 🔗



📄 Blog Post

Introducing: The Bluetooth Low Energy Primer

Bluetooth® technology has been around for more than 20 years. Initially created to allow...

[READ MORE](#) 🔗