Bluetooth mesh networking

From Wikipedia, the free encyclopedia

Bluetooth mesh networking, conceived in 2015,[1] adopted on July 13, 2017[2] is a protocol based upon Bluetooth Low Energy that allows for many-to-many communication over Bluetooth radio.

It has been defined in Mesh Profile Specification[3] and Mesh Model Specification.[4]

Bluetooth mesh networking

Developed by	Bluetooth SIG
Introduced	July 13, 2017
Industry	Lighting, IoT
Physical range	100-1000 meters (depending on mesh relaying configuration)

Contents

Overview

Communication is carried in the messages that may be up to 384 bytes long, when using Segmentation and Reassembly (SAR) mechanism, but most of the messages fit in one segment, that is 11 bytes. Each message starts with an opcode, which may be a single byte (for special messages), 2 bytes (for standard messages), or 3 bytes (for vendor-specific messages).

Every message has a source and a destination address, determining which devices process messages. Devices publish messages to destinations which can be single things / groups of things / everything.

Each message has a sequence number that protects the network against replay attacks.

Each message is encrypted and authenticated. Two keys are used to secure messages: (1) network keys – allocated to a single mesh network, (2) application keys – specific for a given application functionality, e.g. turning the light on vs reconfiguring the light.

Messages have a time to live (TTL). Each time message is received and retransmitted, TTL is decremented which limits the number of "hops", eliminating endless loops.

Bluetooth Mesh is a flood network. It's based on the nodes relaying the messages: every relay node that receives a network packet that authenticates against a known network key that is not in message cache, that has a TTL ≥ 2 can be retransmitted with TTL = TTL - 1. Message cache used to prevent relaying messages recently seen.

Bluetooth Mesh has a layered architecture, with multiple layers as below.

Layer	Functionality
Model Layer	It defines a standard way to exchange application specific messages. For example, a Light Lightness Model defines an interoperable way to control lightness. There are mandatory models, called Foundation Models, defining states and messages needed to manage a mesh network.
Access Layer	It defines mechanism to ensure that data is transmitted and received in the right context of a model and its associated application keys.
Upper Transport Layer	It defines authenticated encryption of access layer packets using an application (or device specific key). It also defines some control messages to manage Friendship or to notify the behavior of node using Heartbeat messages.
Lower Transport Layer	This layer defines a reliable (through a Block Acknowledgement) Segmented transmission upper layer packets, when a complete upper layer packet can't be carried in a single network layer packet. It also defines a mechanism to reassemble segments on the receiver.
Network Layer	This layer defines how transport packets are addressed over network to one or more nodes. It defines relay functionality for forwarding messages by a relay node to extended the range. It handles the network layer authenticated encryption using network key
Bearer Layer	It defines how the network packets are exchanged between nodes. Mesh Profile Specification defines BLE advert bearer and BLE GATT bearer. Mesh Profile defines Proxy Protocol, through which mesh packets can be exchanged via other bearers like TCP/IP

Theoretical limits

It's yet to be determined what are the practical limits of Bluetooth Mesh technology. There are some limits that are built into the specification, though:

Limit for a network	Value	Remarks
Maximum number of nodes	32 767	The limit is 32768 addresses and while a node may occupy more than one address, practical limit is most likely lower
Maximum number of groups	16 384 Number of virtual groups is 2^{128}.	
Maximum number of scenes	65 535	
Maximum number of subnets	4 096	
Maximum TTL	127	

Mesh models

As of version 1.0 of Bluetooth Mesh specification, the following standard models and model groups have been defined:

Foundation models

Foundation models have been defined in the core specification. Two of them are mandatory for all mesh nodes.

- Configuration Server (mandatory)
- Configuration Client
- Health Server (mandatory)
- Health Client

Generic models

- Generic OnOff Server, used to represent devices that do not fit any of the model descriptions defined but support the generic properties of On/Off
- Generic Level Server, keeping the state of an element in a 16-bit signed integer
- Generic Default Transition Time Server, used to represent a default transition time for a variety of devices
- Generic Power OnOff Server & Generic Power OnOff Setup Server, used to represent devices that do not fit any of the model descriptions but support the generic properties of On/Off
- Generic Power Level Server & Generic Power Level Setup Server, including a Generic Power Actual state, a Generic Power Last state, a Generic Power Default state and a Generic Power Range state
- Generic Battery Server, representing a set of four values representing the state of a battery
- Generic Location Server & Generic Location Setup Server, representing location information of an element, either global (Lat/Lon) or local
- Generic User/Admin/Manufacturer/Client Property Server, representing any value to be stored by an element
- Generic OnOff Client & Generic Level Client
- Generic Default Transition Time Client
- Generic Power OnOff Client & Generic Power Level Client
- Generic Battery Client
- Generic Location Client
- Generic Property Client

Sensors

- Sensor Server & Sensor Setup Server, representing a sensor device. Sensor device may be configured to return a measured value periodically or on request; measurement period (cadence) may be configured to be fixed or to change, so that more important value range is being reported faster.
- Sensor Client

Time and scenes

- Time Server & Time Setup Server, allowing for time synchronization in mesh network
- Scene Server & Scene Setup Server, allowing for up to 65535 scenes to be configured and recalled when needed.
- Scheduler Server & Scheduler Setup Server
- Time Client, Scene Client & Scheduler Client

Lighting

- Light Lightness Server & Light Lightness Setup Server, representing a dimmable light source
- Light CTL Server, Light CTL Temperature Server & Light CTL Setup Server, representing a CCT or "tunable white" light source
- Light HSL Server, Light HSL Hue Server, Light HSL Saturation Server & Light HSL Setup Server, representing a light source based on Hue, Saturation, Lightness color representation
- Light xyL Server & Light xyL Setup Server, representing a light source based on modified CIE xyY color space.
- Light LC (Lightness Control) Server & Light LC Setup Server, representing a light control device, able to control Light Lightness model using an occupancy sensor and ambient light sensor. It may be used for light control scenarios like Auto-On, Auto-Off and/or Daylight Harvesting.
- Light Lightness Client, Light CTL Client, Light HSL Client, Light xyL Client & Light LC Client

Provisioning

Provisioning is a process of installing the device into a network. It is a mandatory step to build a Bluetooth Mesh network.

In the provisioning process, a provisioner securely distributes a network key and a unique address space for a device. Provisioning protocol uses P256 Elliptic Curve Diffie-Hellman Key Exchange to create a temporary key to encrypt network key and other information. This provides security from a passive eavesdropper. It also provides various authentication mechanisms to protect network information, from an active eavesdropper who uses Man-In-The-Middle attack, during provisioning process.

A key unique to a device known as "Device Key" is derived from elliptic curve shared secret on provisioner and device during the provisioning process. This device key is used by the provisioner to encrypt messages for that specific device.

Terminology used in Bluetooth mesh networking specification

- Destination: The address to which a message is sent.
- Element: An addressable entity within a device.
- Model: Standardized operation of typical user scenarios.
- Node: A provisioned device.
- Provisioner: A node that can add a device to a mesh network.
- Relay: A node able to retransmit messages.
- Source: The address from which a message is sent.

Implementations

Qualified Bluetooth® mesh implementations

Name	Submitter	Qualification date	QDID	Type
Bluetooth Stack for Embedded Systems - MESH profile[5]	Silvair, Inc.	July 18, 2017	98880	Profile Subsystem
Qualcomm Bluetooth Mesh[6]	Qualcomm Technologies International, Ltd.	July 18, 2017	98856	Profile Subsystem
Silvair Mesh Models[7]	Silvair, Inc.	July 26, 2017	99282	Profile Subsystem
Wireless Gecko Mesh Profile[8]	Silicon Laboratories	September 21, 2017	101318	Profile Subsystem
CYW-MESH 1.0[9]	Cypress Semiconductor Corporation	October 3, 2017	101726	Component (Tested)

References

1. "Get ready for Bluetooth mesh! | Bluetooth Technology Website" (https://blog.bluetooth.com/trashed). *blog.bluetooth.com*. Retrieved 2017-07-06.
2. "Low Energy: Mesh | Bluetooth Technology Website" (https://www.bluetooth.com/what-is-bluetooth-technology/how-it-works/le-mesh). *www.bluetooth.com*. Retrieved 2017-07-18.
3. "Mesh Profile Bluetooth® Specification" (https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=429633) (PDF download). *Bluetooth Technology Website*. 2017-07-13. Retrieved 2017-07-18.
4. "Mesh Model Bluetooth® Specification" (https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=429634) (PDF download). *Bluetooth Technology Website*. 2017-07-13. Retrieved 2017-07-18.
5. "QD ID 98880 | The Official Bluetooth SIG Member Website" (https://www.bluetooth.org/tpg/QLI_viewQDL.cfm?qid=36228). *www.bluetooth.org*. Retrieved 2017-07-19.
6. "QD ID 98856 | The Official Bluetooth SIG Member Website" (https://www.bluetooth.org/tpg/QLI_viewQDL.cfm?qid=34597). *www.bluetooth.org*. Retrieved 2017-07-19.
7. "QD ID 99282 | The Official Bluetooth SIG Member Website" (https://www.bluetooth.org/tpg/QLI_viewQDL.cfm?qid=36327). *www.bluetooth.org*. Retrieved 2017-07-26.

8. "QD ID 101318 | The Official Bluetooth SIG Member Website" (https://www.bluetooth.org/tpg/QLI_viewQDL.cfm?qid=37122). *www.bluetooth.org*. Retrieved 2017-10-03.
9. "QD ID 101726 | The Official Bluetooth SIG Member Website" (https://www.bluetooth.org/tpg/QLI_viewQDL.cfm?qid=35920). *www.bluetooth.org*. Retrieved 2017-10-03.