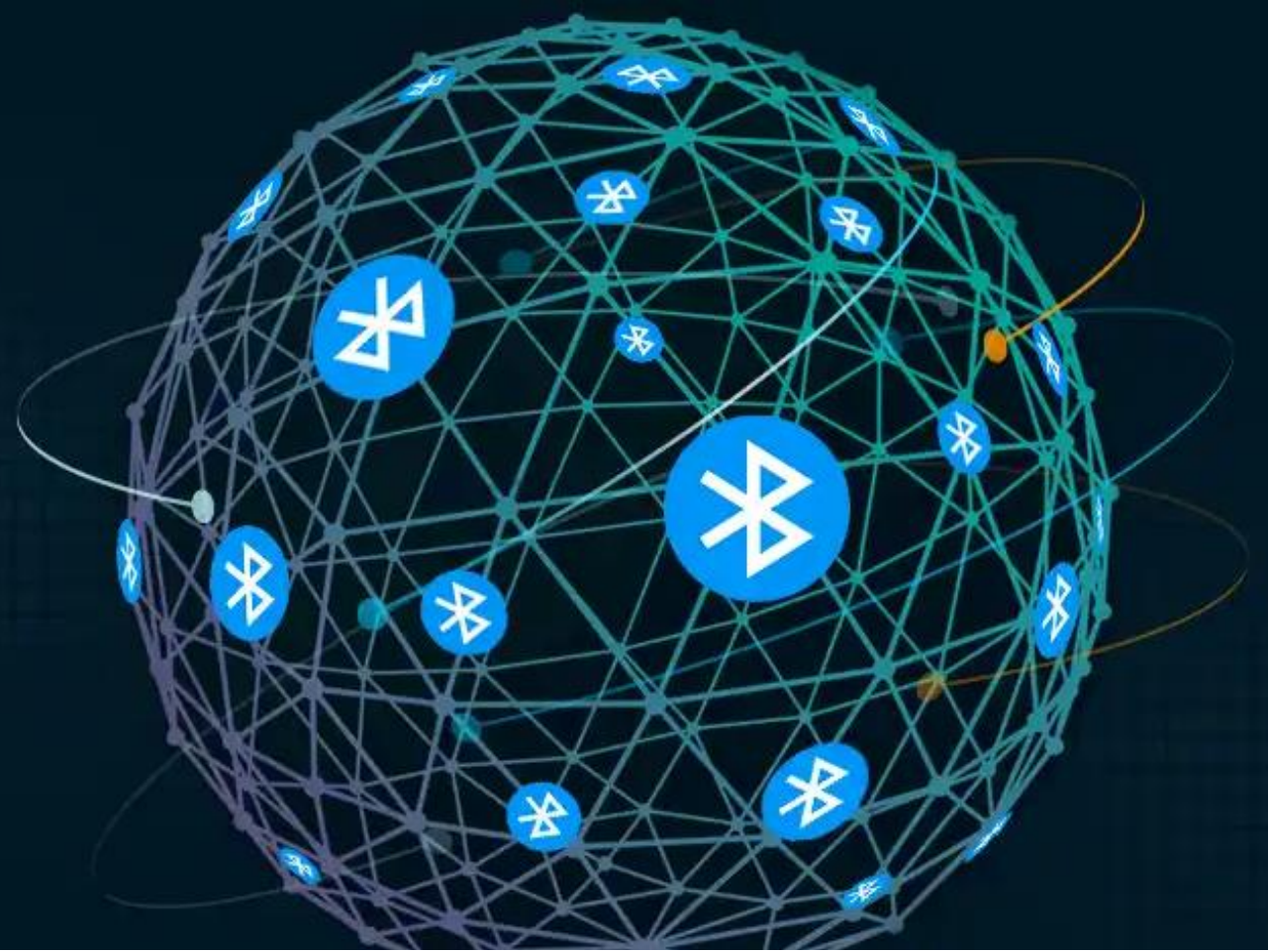




BLUETOOTH MESH



**DEVELOPMENT AND OPTIMIZATION OF A BLUETOOTH
LOW ENERGY (BLE) MESH NETWORK FOR
ENVIRONMENTAL MONITORING AND ALERTING**

Thesis presentation
Mohamed Mahmud Ali
May 22nd 2024



INTRODUCTION

This thesis develops an IoT smart system for integrating devices and sensors, such as air quality, temperature sensors, etc as well as control systems such as lights, gates, fire systems, etc. within a large network. The aim of this thesis is to evaluate the range and data accuracy across various nodes within Bluetooth network.

Challenges:

❑ Data Accuracy and Reliability:

- Sending sensor data over long distances can lead to incorrect data delivery.

❑ Real-Time Data Processing:

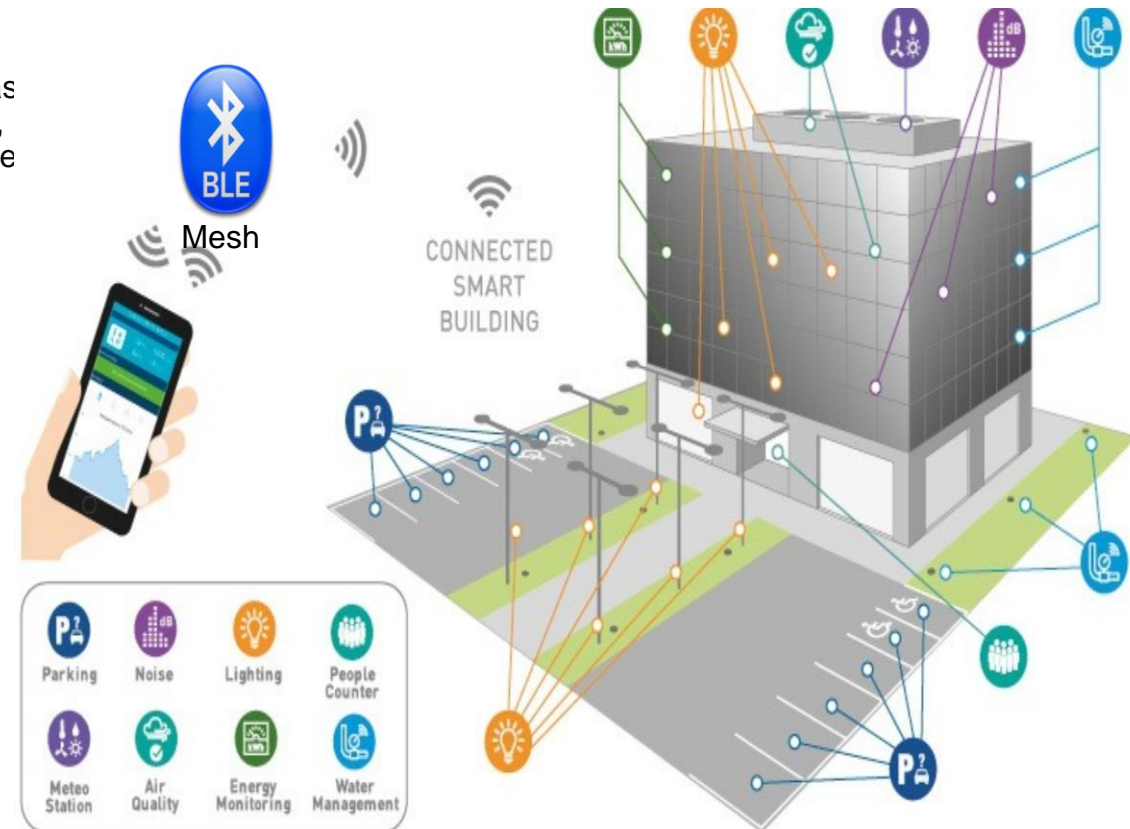
- Delays in data transmission and processing can hinder timely alerting and response actions.
- High volumes of data from numerous sensors require robust and efficient processing systems.

❑ Scalability Issues:

- Expanding monitoring systems to cover larger or more varied areas can be technically and financially challenging.
- Integrating new sensors or expanding network coverage often requires significant system reconfiguration.

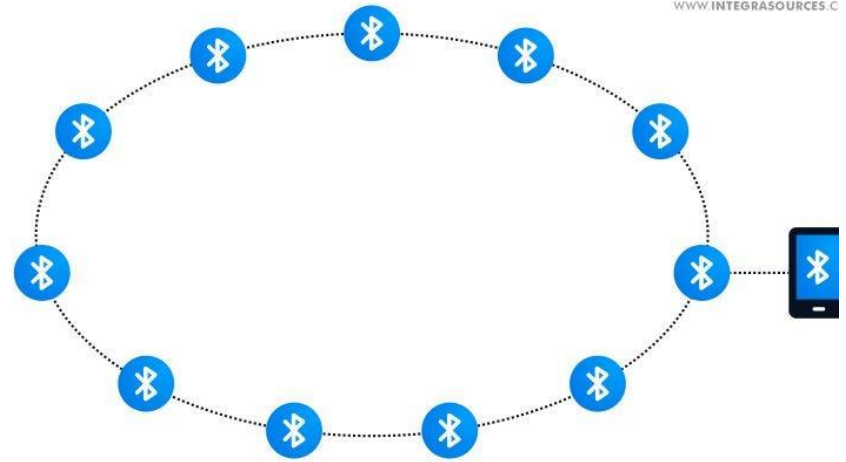
❑ Power Management:

- Remote sensors and devices often have limited power sources, requiring efficient power management to ensure continuous operation.
- Energy consumption becomes a critical issue as the network of sensors expands.



OBJECTIVES

- 1. Create a BLE Mesh Network:**
Design and implement a robust BLE mesh network specifically tailored for environmental monitoring using nRF52 semiconductors microcontrollers.
- 2. Ensure Data Reliability**
Monitor and record the forwarding of messages by each relay node to adjacent nodes within the network.
- 3. Optimize Network Performance:**
Enhance scalability and power efficiency to ensure the network can handle increasing loads with minimal energy consumption.
- 4. Test and Validate the System:**
Conduct comprehensive testing in real-world environments to validate the performance and effectiveness of the BLE mesh network.



nRF52

SOLUTION

Development of a BLE Mesh Network: Suitable specifically for monitoring, alerting and control, leveraging the unique properties of Bluetooth Low Energy technology.

Scalability:

- Supports up to 32,767 nodes /sensors within the network.
- Ideal for complex and extensive environmental monitoring systems.

Reliability

- Data can be relayed across multiple paths, enhancing the network's robustness and reducing the risk of data loss.

Energy Efficiency:

- Optimized for minimal power usage, significantly prolonging battery life in deployed sensor devices.
- Reduces maintenance frequency and operational costs.

Wide Area Coverage:

- Capable of covering large geographical areas.
- Ensures comprehensive environmental data collection across diverse locations.

WHAT IS BLUETOOTH MESH ?

Bluetooth mesh is a network protocol designed for creating large-scale device networks. It is ideal for building automation, sensor networks, and other IoT (Internet of Things) solutions where hundreds or thousands of devices need to reliably and securely communicate with one another.

DIFFERENCES:

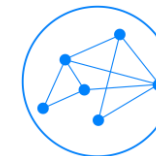
- **Bluetooth classic**
 - exchange data over short distances using radio waves 2.4GHz (10 meters range).
 - Topology : point-to-point or at best star
- **BLE**
 - Compared to Classic Bluetooth, Bluetooth Low Energy is intended to provide considerably reduced power consumption and cost while maintaining a similar communication range. (10-meter range).
 - Topology: point-to-point or at best star
- **BLE mesh**
 - is a protocol build on BLE stack to connect and communicate in mesh network (The more nodes in network, more range).
 - Topology : flooded mesh



Bluetooth
Classic



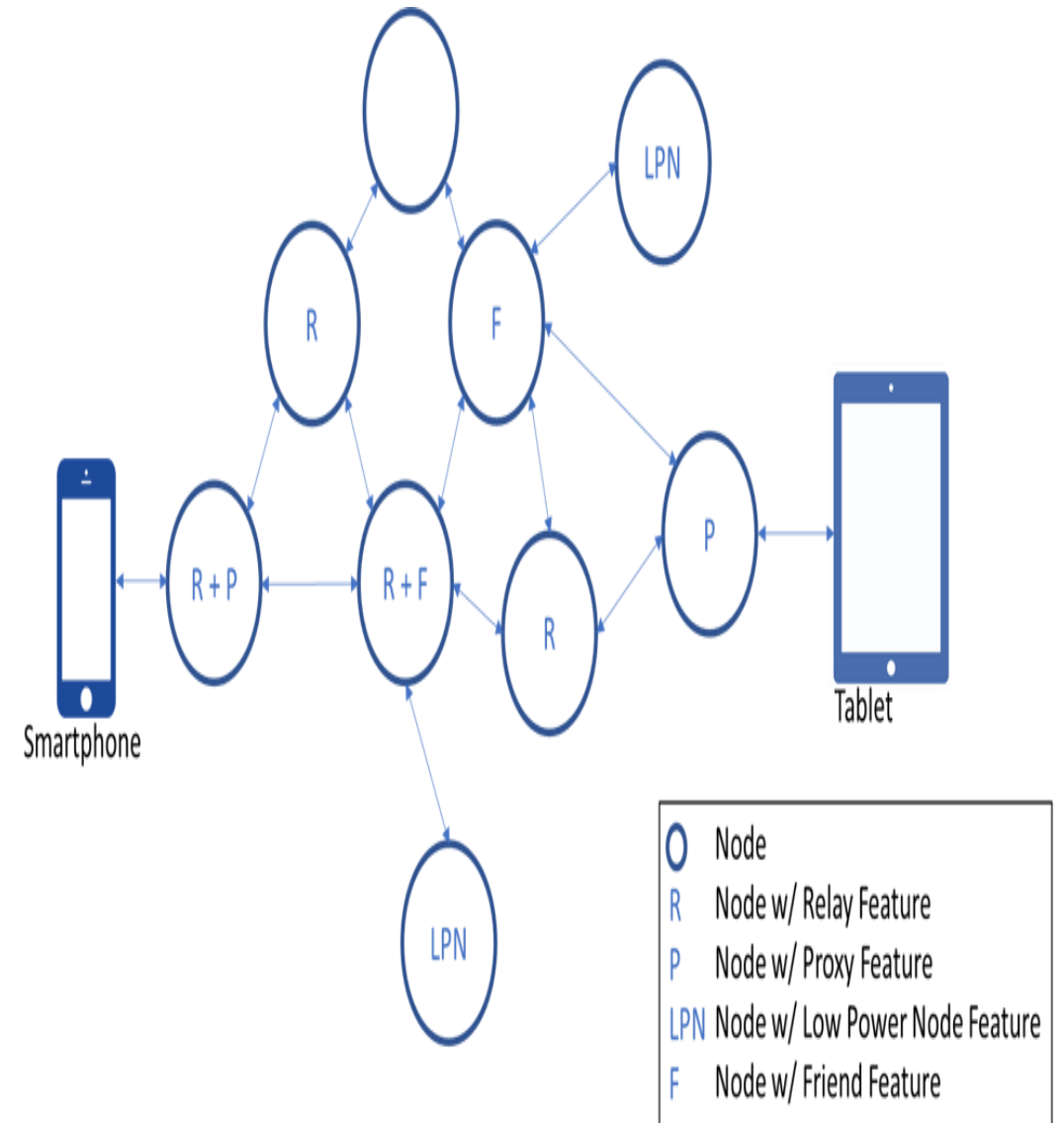
Bluetooth low
energy



BLE-mesh

FEATURES AND ROLES

- **Relay Node**
 - Enables a node to relay messages over the advertising bearer.
- **Low Power Node (LPN)**
 - conserves energy by allowing devices to sleep, waking only to check for messages with a Friend node.
- **Friend Node**
 - will listen for any messages that are relayed in the network and intended for an associated Low-Power Node.
- **Proxy Node**
 - The Proxy feature allows a node to relay messages between the GATT (General ATtribute) and advertising bearers.
- **Provisioner**
 - is a device that add unprovisioned nodes in the network



RELAYING

Relay devices retransmit messages

- Maximum of 127 hops
- Enough to relay across an enormous physical area

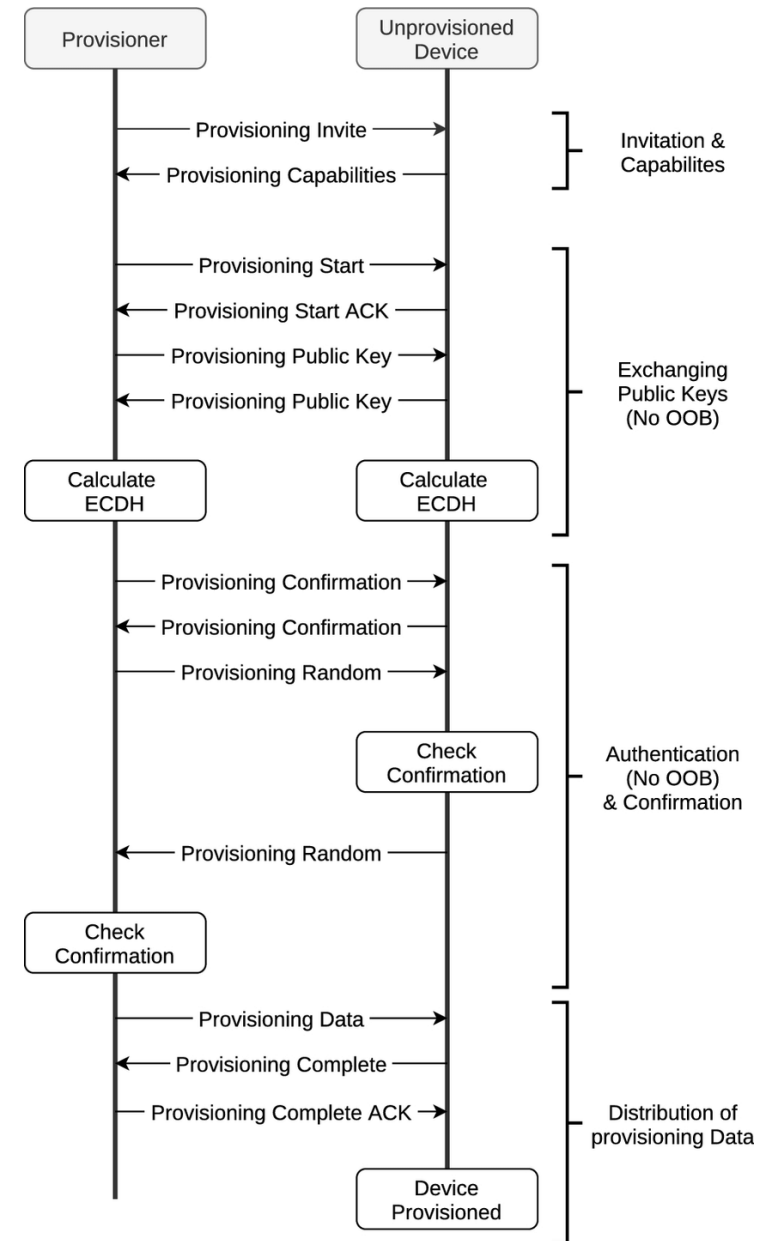
Managed flooding

- **Retransmit the message to all other devices in range**
- **All packets include a time-to-live (TTL)**
- **Message cache to reduce overhead**

PROVISIONING

Provision is the process of providing new devices in the Bluetooth mesh network with the information they need to join a network. To become a node and participate in the Bluetooth mesh communication, each device must be provisioned. Provision following two roles:

- The provisioner represents the network owner, and is responsible for adding new nodes to the mesh network.
- The provisionee is the device that gets added to the network through the Provisioning process. Before the provisioning process starts, the provisionee is an unprovisioned device.



SECURITY

Network layer

- Every packet is encrypted and authenticated
- Sequence numbers to avoid replay attacks
- Man-in-the middle protection
- Protection against trash-can attacks
- Security keys are refreshed on regular basis

Access layer

- Separation into network keys, application keys and device keys.

ENCRYPTED DATA

nRF Sniffer for Bluetooth LE COM14

File Edit View Go Capture Analyze Statistics Telegraphy Wireless Tools Help

ismesh

No.	Time	Delta	Source	Destination	Protocol	Length	Info
2202	2024-05-11 16:36:21.060200	0.000000	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2203	2024-05-11 16:36:21.060224	0.000036	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2204	2024-05-11 16:36:21.060610	0.000386	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2295	2024-05-11 16:36:21.099927	0.021317	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2296	2024-05-11 16:36:21.099933	0.000036	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2297	2024-05-11 16:36:21.099939	0.000036	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2301	2024-05-11 16:36:21.114784	0.023805	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2302	2024-05-11 16:36:21.115170	0.000386	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2303	2024-05-11 16:36:21.115556	0.000386	2e:3e:02:41:c8:10	Broadcast	BT Mesh	54	ADV_NOTIFICATION
2626	2024-05-11 16:36:28.986662	7.331806	7e:c3:a0:7e:c3:a1	Broadcast	BT Mesh	65	ADV_NOTIFICATION [Formatted Packet]
4430	2024-05-11 16:36:33.152624	0.646810	17:e2:a6:7e:c3:a1	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4440	2024-05-11 16:36:33.179535	0.020883	17:e2:a6:7e:c3:a1	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4441	2024-05-11 16:36:33.179953	0.000418	17:e2:a6:7e:c3:a1	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4442	2024-05-11 16:36:33.180372	0.000418	17:e2:a6:7e:c3:a1	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4446	2024-05-11 16:36:33.208063	0.027091	17:e2:a6:7e:c3:a1	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4447	2024-05-11 16:36:33.208481	0.000418	17:e2:a6:7e:c3:a1	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4448	2024-05-11 16:36:33.208900	0.000419	17:e2:a6:7e:c3:a1	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4680	2024-05-11 16:36:34.269946	0.368485	08:c1:b8:c8:c9:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4689	2024-05-11 16:36:34.269723	0.000418	08:c1:b8:c8:c9:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4628	2024-05-11 16:36:34.195540	0.025777	08:c1:b8:c8:c9:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
4621	2024-05-11 16:36:34.195959	0.000419	08:c1:b8:c8:c9:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5037	2024-05-11 16:36:36.639223	2.397234	3e:8a:9f:21:10:74	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5034	2024-05-11 16:36:36.616490	0.023197	3e:8a:9f:21:10:74	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5035	2024-05-11 16:36:36.616908	0.000418	3e:8a:9f:21:10:74	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5036	2024-05-11 16:36:36.617326	0.000418	3e:8a:9f:21:10:74	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5214	2024-05-11 16:36:37.320400	0.365154	21:c1:0c:1e:00:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5207	2024-05-11 16:36:37.046930	0.024448	21:c1:0c:1e:00:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5228	2024-05-11 16:36:37.067375	0.000419	21:c1:0c:1e:00:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5229	2024-05-11 16:36:37.067793	0.000418	21:c1:0c:1e:00:90	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5356	2024-05-11 16:36:38.565888	0.355111	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5367	2024-05-11 16:36:38.567395	0.000419	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5368	2024-05-11 16:36:38.567722	0.000417	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5370	2024-05-11 16:36:38.593905	0.025363	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5379	2024-05-11 16:36:38.593963	0.000418	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5380	2024-05-11 16:36:38.593922	0.000419	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5384	2024-05-11 16:36:38.615080	0.025081	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5385	2024-05-11 16:36:38.615491	0.000418	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
5386	2024-05-11 16:36:38.615909	0.000418	39:09:18:97:fe:62	Broadcast	BT Mesh	58	ADV_NOTIFICATION
11933	2024-05-11 16:37:14.385315	35.485276	Apple_3a:10:19	Broadcast	BT Mesh	49	ADV_NOTIFICATION [Formatted Packet]
20986	2024-05-11 16:38:01.422322	47.317087	48:c9:c4:1ae:28	Broadcast	BT Mesh	49	ADV_NOTIFICATION [Formatted Packet]
25020	2024-05-11 16:38:25.918825	24.489513	67:9c:1f:9:10:c1	Broadcast	BT Mesh	49	ADV_NOTIFICATION [Formatted Packet]
25028	2024-05-11 16:38:29.268959	3.470102	14:c5:c1:08:44:60	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25021	2024-05-11 16:38:29.268324	0.000418	14:c5:c1:08:44:60	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25022	2024-05-11 16:38:29.268172	0.000418	14:c5:c1:08:44:60	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25085	2024-05-11 16:38:30.395643	3.045697	39:08:b6:d5:cccd	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25086	2024-05-11 16:38:30.396847	0.000418	39:08:b6:d5:cccd	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25087	2024-05-11 16:38:30.397266	0.000419	39:08:b6:d5:cccd	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25014	2024-05-11 16:38:30.422752	0.025486	39:08:b6:d5:cccd	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25015	2024-05-11 16:38:30.423171	0.000419	39:08:b6:d5:cccd	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25016	2024-05-11 16:38:30.423588	0.000417	39:08:b6:d5:cccd	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25047	2024-05-11 16:38:31.175848	0.352212	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25088	2024-05-11 16:38:31.176279	0.000419	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25092	2024-05-11 16:38:31.199911	0.020322	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25093	2024-05-11 16:38:31.198038	0.000419	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25094	2024-05-11 16:38:31.198074	0.000418	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25098	2024-05-11 16:38:31.424248	0.021600	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
25099	2024-05-11 16:38:31.422847	0.000419	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
26080	2024-05-11 16:38:31.423265	0.000418	1e:cf:b8:ff:95:7d	Broadcast	BT Mesh	58	ADV_NOTIFICATION
26159	2024-05-11 16:38:31.268959	0.362193	1e:1d:af:18:11:41	Broadcast	BT Mesh	58	ADV_NOTIFICATION
26159	2024-05-11 16:38:31.268736	0.000418	1e:1d:af:18:11:41	Broadcast	BT Mesh	58	ADV_NOTIFICATION
26160	2024-05-11 16:38:31.268794	0.000418	1e:1d:af:18:11:41	Broadcast	BT Mesh	58	ADV_NOTIFICATION
31590	2024-05-11 16:39:05.489513	33.180179	Apple_3a:10:19	Broadcast	BT Mesh	49	ADV_NOTIFICATION [Formatted Packet]
33695	2024-05-11 16:39:17.285955	11.790142	Apple_3b:10:19	Broadcast	BT Mesh	177	ADV_NOTIFICATION

0010 89 d6 be 89 84 E2 20 d1 cc c5 b6 98 30 2a 52 ... 8

Length: 82 (btcommen_en_advertising_length), 1 byte

Packets: 37634 - Displayed: 63 (2%)

DECRYPTED DATA

No. Time Delta Source Destination Protocol Length Info

20452	2024-05-10 20:41:17.002380	0.004934	23:46:af:aa:ad:c0	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20453	2024-05-10 20:41:17.002807	0.000419	23:46:af:aa:ad:c0	Broadcast	BT Mesh	58	ADV_NOTIFICATION
20452	2024-05-10 20:41:17.002855	1.000230	00:00:00:00:00:00	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20504	2024-05-10 20:41:18.074018	0.222755	43:79:3f:17:13:1b	Broadcast	BT Mesh	40	ADV_ID
20504	2024-05-10 20:41:18.076020	0.708510	00:42:12:7:08:03:04	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20505	2024-05-10 20:41:18.703746	0.000418	00:42:12:7:08:03:04	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20507	2024-05-10 20:41:18.080244	0.010398	2e:89:38:10:af:48	Broadcast	BT Mesh	58	ADV_NOTIFICATION
20508	2024-05-10 20:41:18.080662	0.000418	2e:89:38:10:af:48	Broadcast	BT Mesh	58	ADV_NOTIFICATION
20509	2024-05-10 20:41:18.081060	0.000739	2e:89:38:10:af:48	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20570	2024-05-10 20:41:18.012170	0.000239	13:6d:0f:7ad:5b	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20571	2024-05-10 20:41:18.012707	0.000517	11:72:5f:6d:66:c3	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20572	2024-05-10 20:41:18.013395	0.000418	11:72:5f:6d:66:c3	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20703	2024-05-10 20:41:22.064520	3.040807	04:2d:7aa:2:5ac:3a	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20745	2024-05-10 20:41:23.059730	1.000770	1f:10:1e:38:f3:92	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20793	2024-05-10 20:41:24.046543	0.000973	Cisq_54:58:01	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20845	2024-05-10 20:41:25.065744	1.021206	94:08:41:52:2d:89	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20823	2024-05-10 20:41:25.099461	0.020737	2e:aa:0b:2e:62:37	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20847	2024-05-10 20:41:28.077130	0.000957	04:0e:36:52:2f:16	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20812	2024-05-10 20:41:28.913303	1.038065	31:02:03:07:0e:c0	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20813	2024-05-10 20:41:28.913621	0.000418	31:02:03:07:0e:c0	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20814	2024-05-10 20:41:28.914040	0.000419	31:02:03:07:0e:c0	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20860	2024-05-10 20:41:30.048680	0.000820	28:01:9e:08:0e:20	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
21030	2024-05-10 20:41:32.049400	0.009739	1e:7f:93:1e:21:00	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
21030	2024-05-10 20:41:32.051200	0.000418	1e:7f:93:1e:21:00	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
21037	2024-05-10 20:41:32.013719	0.000418	01:0f:56:15:10:a2	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
21032	2024-05-10 20:41:34.043690	2.030541	10:f1:7e:05:05:40	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
21027	2024-05-10 20:41:35.520839	0.007179	3e:54:34:c1:cc:c0	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
21032	2024-05-10 20:41:35.529456	0.000417	3e:54:34:c1:cc:c0	Broadcast	BT Mesh	58	ADV_NOTIFICATION
21029	2024-05-10 20:41:35.529074	0.000418	3e:54:34:c1:cc:c0	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20975	2024-05-10 20:41:36.955700	1.021096	1e:04:0f:0c:46:10	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20955	2024-05-10 20:41:36.962896	0.000418	1e:04:0f:0c:46:10	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20883	2024-05-10 20:41:42.000919	1.022113	1d:08:3d:58:02:a1	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20864	2024-05-10 20:41:44.013671	1.024552	10:17:11:41:03:63	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20720	2024-05-10 20:41:45.016173	1.002920	05:1b:3:0:0:0	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20955	2024-05-10 20:41:48.016304	3.000031	25:08:08:6d:94:24	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20954	2024-05-10 20:41:48.016703	0.000419	25:08:08:6d:94:24	Broadcast	BT Mesh	58	ADV_NOTIFICATION
20860	2024-05-10 20:41:48.044000	0.020265	10:0f:05:0e:0a:7c	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20929	2024-05-10 20:41:58.040497	0.000419	10:0f:05:0e:0a:7c	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20820	2024-05-10 20:41:52.057900	0.023544	16:15:11:03:1e:02	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20865	2024-05-10 20:41:53.048091	0.000409	39:0e:20:30:94:60	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20804	2024-05-10 20:41:54.070495	1.030044	13:fc:03:03:08:56	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20804	2024-05-10 20:41:54.070923	0.000419	13:fc:03:03:08:56	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20807	2024-05-10 20:41:54.071629	0.000418	13:fc:03:03:08:56	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20843	2024-05-10 20:41:58.068219	0.023472	3a:39:4e:3d:2f:04	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20834	2024-05-10 20:41:59.080391	0.009572	1e:40:71:0f:08:00	Broadcast	BT Mesh	58	ADV_NOTIFICATION
20423	2024-05-10 20:41:59.126633	1.037022	05:16:3c:23:05:44:30	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20424	2024-05-10 20:41:59.124852	0.000419	05:16:3c:23:05:44:30	Broadcast	BT Mesh	58	Generic OffSet Set Unacknowledged
20425	2024-05-10 20:41:59.124470						

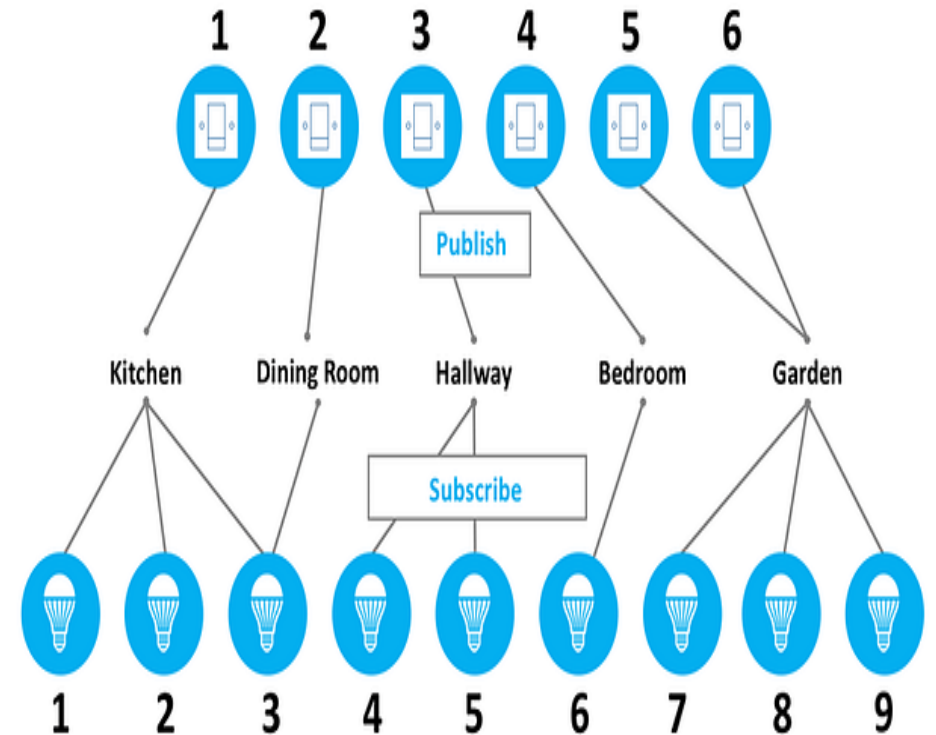
PUBLISH AND SUBSCRIBE

Publishing

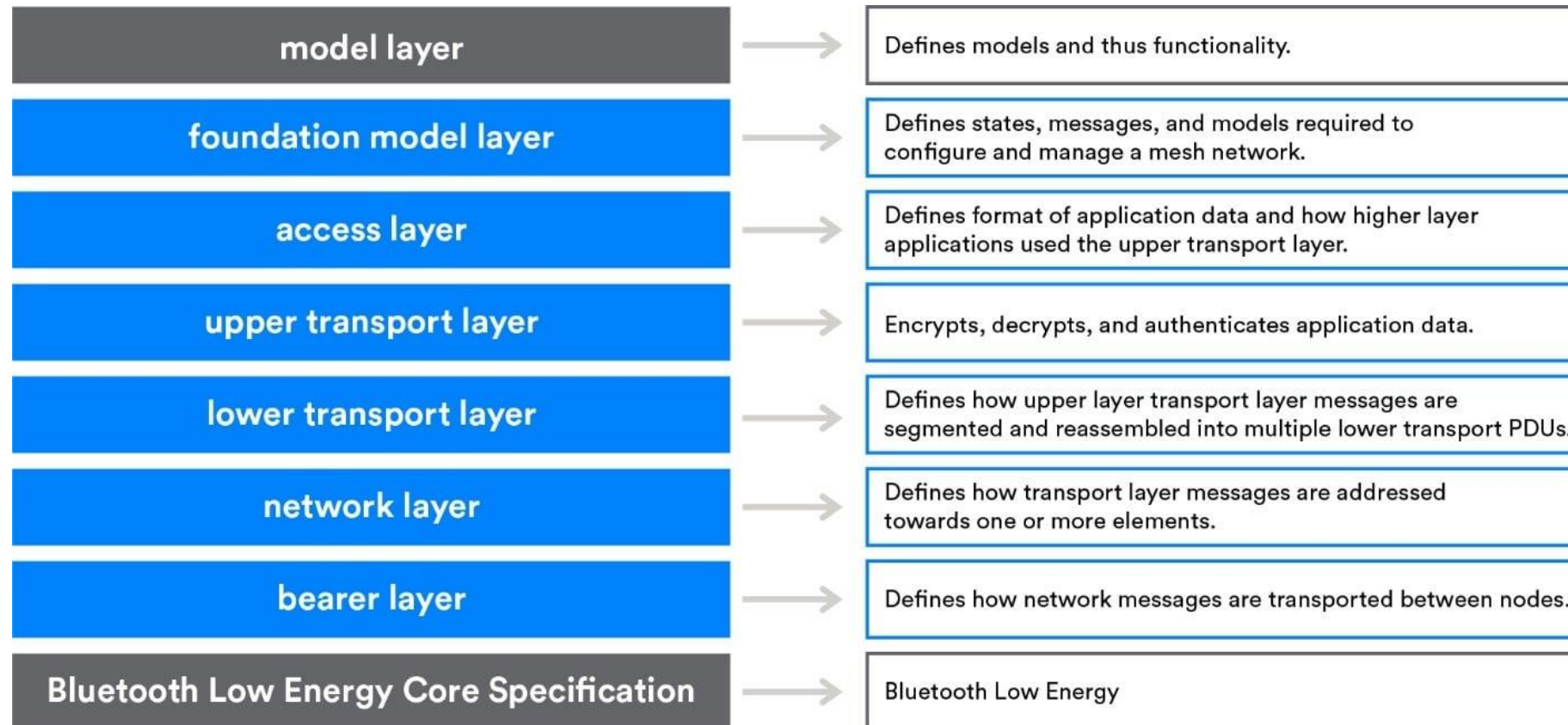
- Devices may send messages to addresses (group), all the other devices that subscribed to that address will receive a copy of it, process it and react in some way.

Subscribing

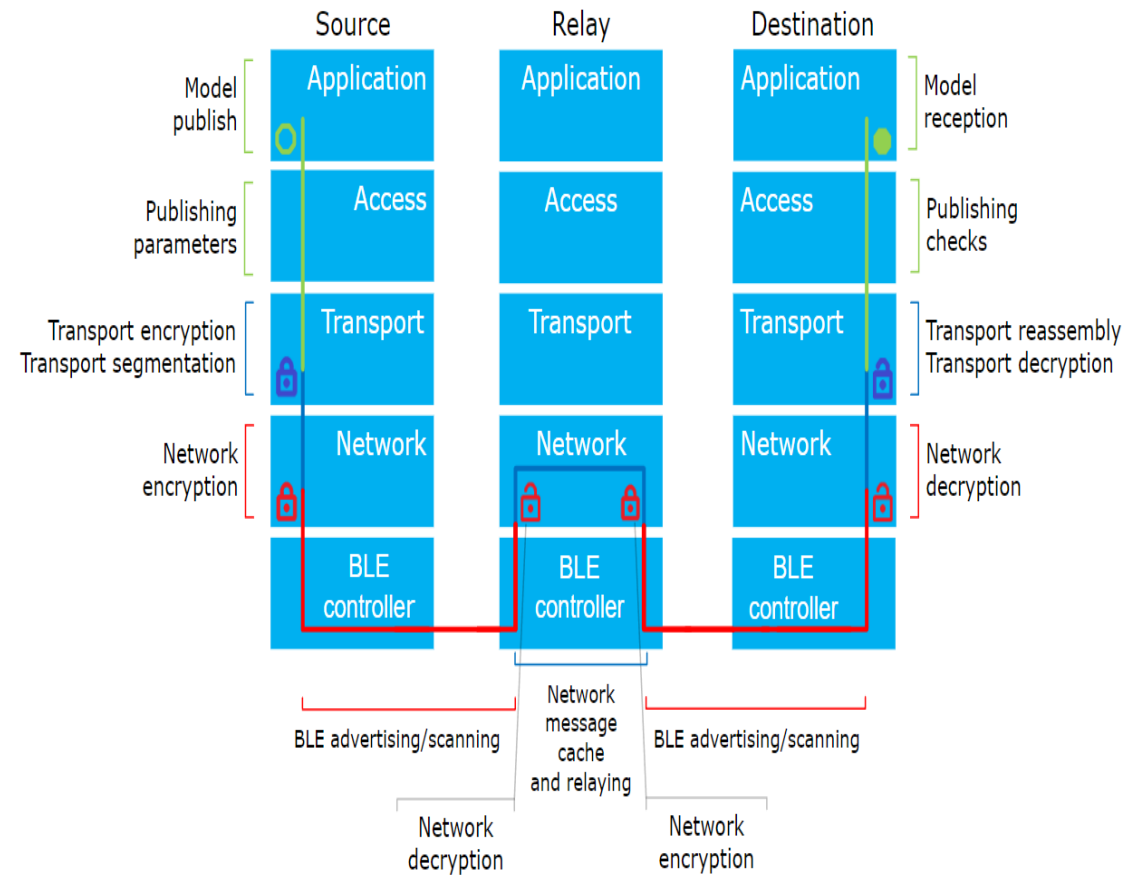
- Devices can be configured to receive messages which were sent to addresses (group) by other device.



BLE MESH ARCHITECTURE



BLE MESH NETWORK DATA FLOW



MODELS

The applications are implemented as a set of mesh models. The Bluetooth SIG defines some generic and reusable models in the Bluetooth Mesh model specification, but vendors are free to define their own models.

generics

- generic onoff client
- generic onoff server
- generic level client
- generic level server
- generic default transition time client
- generic default transition time server
- generic power onoff client
- generic power onoff server
- generic power onoff setup server
- generic power level client
- generic power level server
- generic power level setup server
- generic battery client
- generic battery server
- generic location client
- generic location server
- generic location setup server
- generic admin property server
- generic manufacturer property server
- generic user property server
- generic admin property server
- generic property client

sensors

- sensor client
- sensor server
- sensor setup server

time and scenes

- time client
- time server
- time setup server
- scene client
- scene server
- scene setup server
- scheduler client
- scheduler server
- scheduler setup server

lighting

- light lightness client
- light lightness server
- lightness setup server
- light CTL client
- light CTL server
- light CTL setup server
- light HSL client
- light HSL server
- light HSL setup server
- light xyL client
- light xyL server
- light xyL setup server
- light LC client
- light LC server
- light LC setup server

EXPERIMENT SETUP

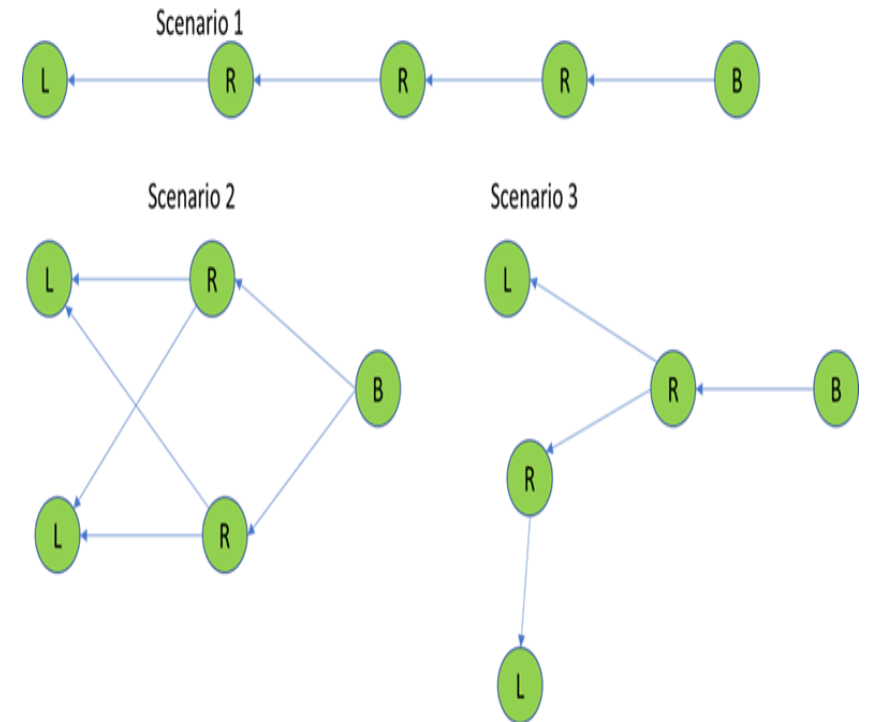
System contains 3 roles:

B: broadcaster

R: Relay

L: Listener

- Broadcaster initiate a sequence of messages to all nodes.
- Relays receives and calculate PLR (Packet Lose Ratio) and forward them to adjacent nodes.
- Listeners will record these messages and calculate PLR also.



SCENARIOS TESTS

Links between the nodes will be 15 and 30 meters.

- **Scenario 1:**

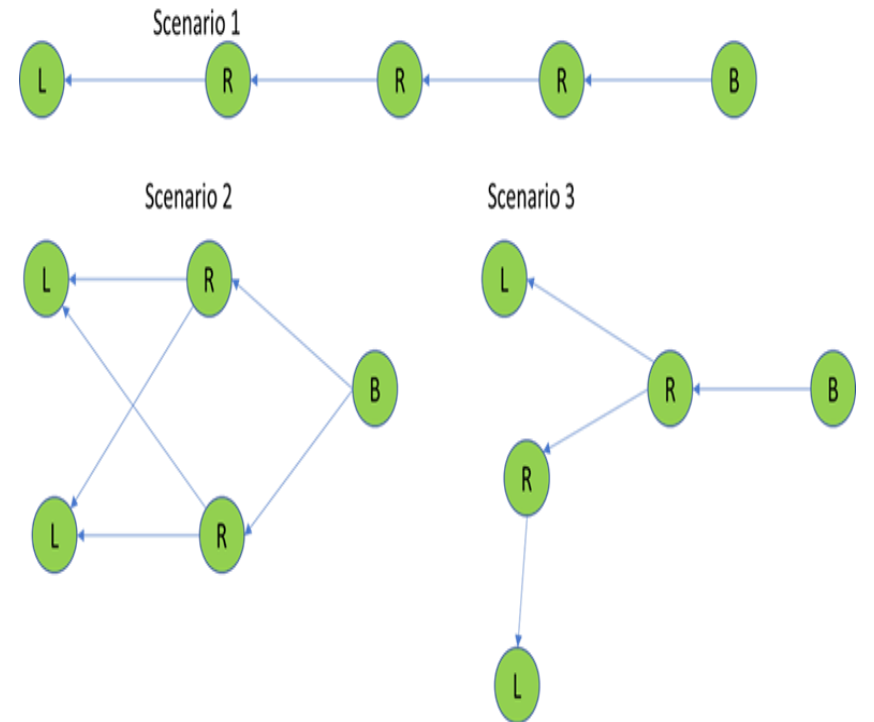
The broadcaster will send 60, 120, and 600 messages over a 60-second period. Each node will record the Packet Loss Ratio (PLR).

- **Scenario 2:**

Two relay nodes will be positioned centrally to facilitate message forwarding to the listeners.

- **Scenario 3:**

One listener will receive messages through a single hop, while another listener will receive messages through two hops.



PLR CALCULATION

```
static int gen_onoff_set_unack(const struct bt_mesh_model *model,
                             struct bt_mesh_msg_ctx *ctx,
                             struct net_buf_simple *buf)
{
    if (buf->len < 4) {
        printk("Buffer too short\n");
        return -EINVAL;
    }
    uint8_t val = net_buf_simple_pull_u8(buf);
    uint16_t seq = net_buf_simple_pull_le16(buf);
    uint8_t tid = net_buf_simple_pull_u8(buf);

    int32_t trans = 0;
    int32_t delay = 0;
    printk("Receiving onoff:%d, tid:%d ,src:%d ,seq:%u \n", val, tid, ctx->addr ,seq);

    PLR(seq);
    /*if (ctx->recv_ttl < 1) {
        rebroadcast(model, ctx, seq, ctx->recv_ttl);
    }*/

    if (buf->len) {
        trans = model_time_decode(net_buf_simple_pull_u8(buf));
        delay = net_buf_simple_pull_u8(buf) * 5;
    }

    /* Only perform change if the message wasn't a duplicate and the
     * value is different.
     */
    if (tid == onoff.tid && ctx->addr == onoff.src) {
        /* Duplicate */
        return 0;
    }

    if (val == onoff.val) {
        /* No change */
        return 0;
    }
}
```

```
static int PLR(uint16_t seq_num) {
    // Static variables to keep track of sequence and statistics
    static uint16_t last_seq_num = 0;
    static uint16_t first_seq_seq = 0;
    static bool is_first_packet = true;
    static uint16_t total_missed_packets = 0;
    static uint16_t total_received_packets = 0;

    // Handling the first packet received
    if (is_first_packet) {
        first_seq_seq = seq_num;
        last_seq_num = seq_num; // Initialize the last sequence number with the first received sequence
        is_first_packet = false; // Mark that the first packet has now been received
        total_received_packets++; // Start counting packets with the first one received
    } else {
        // Normal handling for all subsequent packets
        if (seq_num > last_seq_num) {
            // Calculating missed packets if there's a gap
            total_missed_packets += (seq_num - last_seq_num - 1);
            last_seq_num = seq_num; // Update the last received sequence number
            total_received_packets++; // Increment the total received packets count
        } else if (seq_num == last_seq_num) {
            // Handle duplicate packet
            printf("Duplicate packet received, seq_num: %u\n", seq_num);
        } else {
            // Handle out of order packet
            printf("Out of order packet received, seq_num: %u, last_seq_num: %u\n", seq_num, last_seq_num);
        }
    }

    // Calculate Packet Loss Ratio (PLR)
    uint32_t total_packets_considered = total_received_packets + total_missed_packets;
    uint32_t plr = (total_missed_packets * 10000) / total_packets_considered;

    printk("First received packet: %u\n", first_seq_seq);
    printk("Total missed packets: %u\n", total_missed_packets);
    printk("Total received packets: %u\n", total_received_packets);
    printk("Received seq_num %u, PLR: %u.%02u%%\n", seq_num, plr / 100, plr % 100);
    printk(".....\n");

    return 0;
}
```

RESULTS

Scenario1:

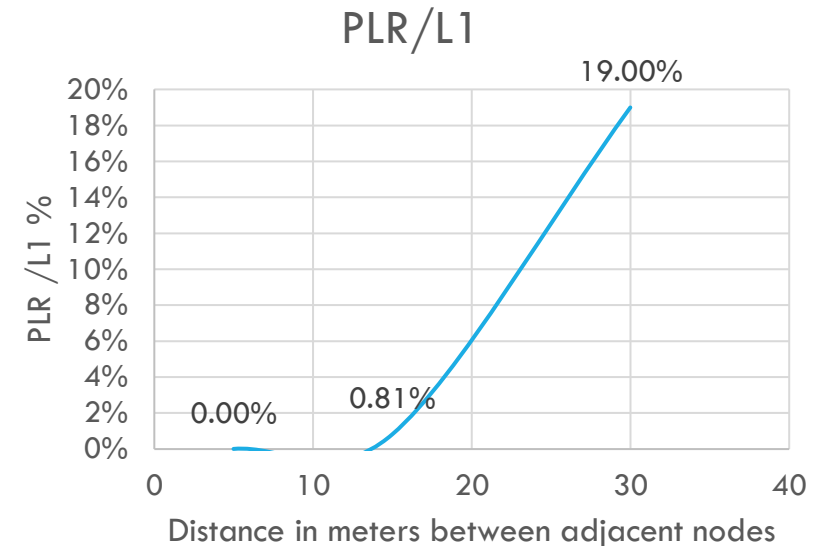
•At ~15 meters:

- Packet Loss Ratio (PLR) was minimal across all message sizes (60, 120, 600), with nearly zero loss at the relay nodes and minor loss at the listener.
- Received Signal Strength Indicator (RSSI) ranged from -60dBm to -53dBm, indicating strong signal strength.

•At ~30 meters:

- PLR increased notably, especially with higher message counts (120, 600), showing increased loss primarily at the listener.
- RSSI decreased to as low as -81dBm, suggesting weaker signals at greater distances.

Distance	payload	PLR /L	PLR /R3	PLR /R2	PLR /R1	RSSI
~15m	60	1.44%	0%	0%	0%	~-60dBm
	120	0.83%	0%	0%	0%	~-52dBm
	600	0.17%	0.34%	0.52%	0.0%	~-53dBm
~30m	60	16.3%	11.7%	3.27%	3.27%	~-68dBm
	120	26%	4.68%	0%	0%	~-81dBm
	600	14.74%	5.16%	5.16%	1.06%	~-77dBm



RESULTS

Scenario2:

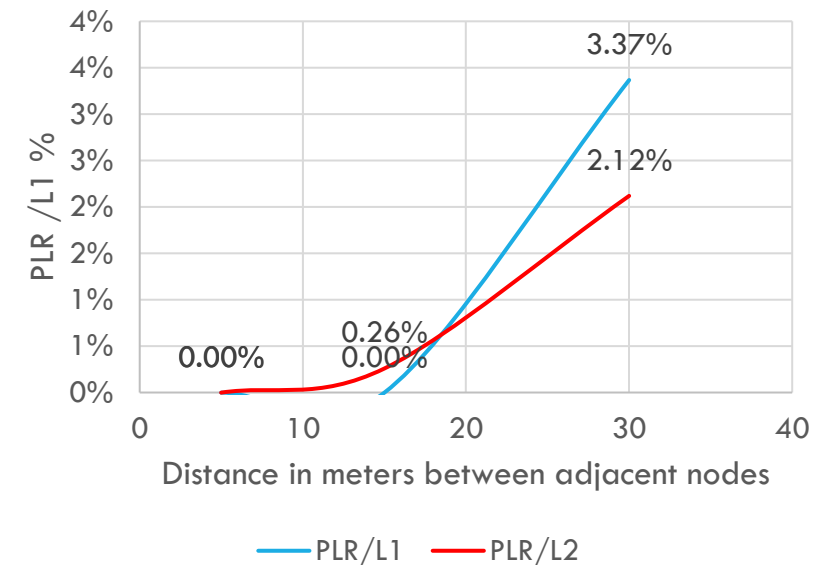
• At 15 meters:

- Very low to zero PLR across all nodes and message sizes, with excellent RSSI values around -51dBm to -54dBm.

• At 30 meters:

- PLR varied, with higher message sizes showing increased loss, particularly notable in messages directed to listeners.
- RSSI showed some decrease but remained relatively strong, suggesting that the double relay setup helps maintain signal integrity over greater distances.

Distance	payload	PLR /L1	PLR /L2	PLR /R2	PLR /R1	RSSI
15m	60	0%	0%	0%	0%	~-51dBm
	120	1.3%	0.78%	6.3%	3.9%	~-57dBm
	600	0%	0%	0.16%	0.16%	~-54dBm
30m	60	0	0%	4.76%	0%	~-57dBm
	120	8.13	0.80%	0%	0%	~-60dBm
	600	1.98%	1.32%	1.48%	0.49%	~-64dBm



RESULTS

Scenario 3:

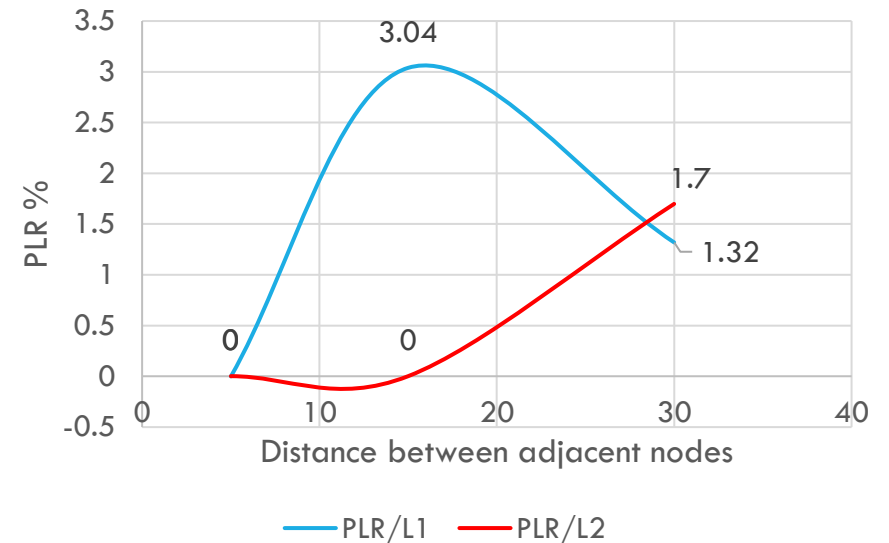
• At 15 meters:

- Higher PLR for the listener requiring more hops (L1), especially as message size increased, indicating that additional hops can lead to increased packet loss.
- RSSI was slightly lower but still within a good range, averaging around -57dBm to -63dBm.

• At 30 meters:

- Significant PLR increase for both listeners as message size increased, particularly for L2.
- RSSI values showed more substantial drops, highlighting challenges with signal strength at longer distances and multiple hops.

Distance	payload	PLR /L1	PLR /L2	PLR /R2	PLR /R1	RSSI
15m	60	4.54	0	0	0	~-63dBm
	120	2.15	0	0	0	~-57dBm
	600	2.43	0	0	0	~-58dBm
30m	60	0	0	3.27	0	~-58dBm
	120	0.78	0	0.78	0	~-55dBm
	600	3.18	5.20	5.03	0.50	~-63dBm



Q&A



THANK YOU

