ESL solution introduction

(Connecting your ESL to IOT cloud promptly)

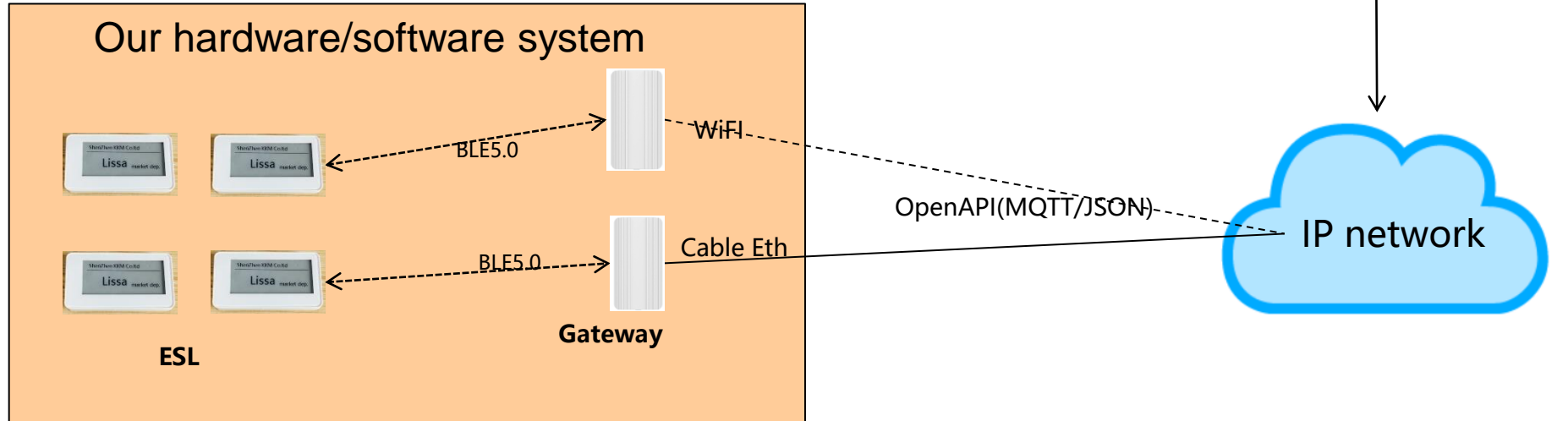# Outline

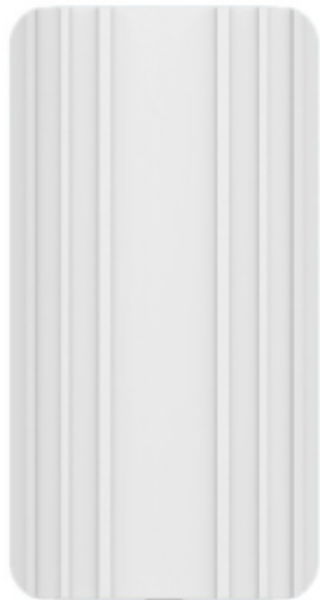> **Introduction**

   - ✓ **Architecture**
   - ✓ **ESL and Gateway**

> **Solution Feature**

Software/hardware Architecture

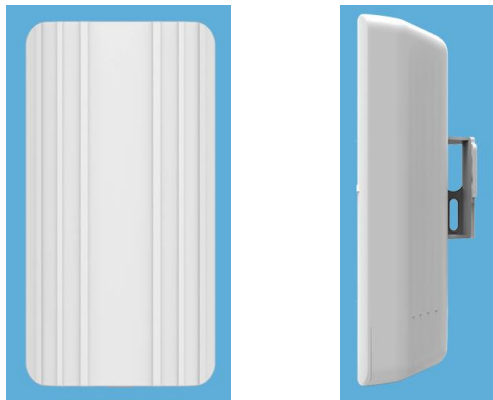ESL/Gateway using **MQTT/JSON open API** for third part to easy integrate it.

Customer Inventory System

Subscribe /Publish

MQTT Broker

Our hardware/software system

BLE5.0

WiFi

OpenAPI(MQTT/JSON)

IP network

BLE5.0

Cable Eth

Gateway

ESL

ESL Gateway

- ➢ **Water proof-IP54/Sun proof（Industry only）**
- ➢ **BLE5.0 Long distance Feature（Industry only）**
- ➢ **BLE distance: > 200 meters in open space**
- ➢ **Self-organized for ESL device**
- ➢ **Open protocol: MQTT to cloud**
- ➢ **Configuration: Web portal**
- ➢ **Transmit: ETH/Wifi/Wifi-hopping**
- ➢ **Power: 5V DC and ETH PoE(802.3af)**
- ➢ **Base on latest OpenWrt 18.29**
- ➢ **BLE chip: TI CC2640R2F**

ESL Gateway Specifications

Item	Description
Power	• 802.3af PoE • DC 5V
Manage ESL number	> 200 ESL
Wireless distance	• BLE5.0: > 200 meters • BLE4.0/4.1/4.2 > 100 meters (depends on environment)
Transmitting way	• ETH RJ45 • WiFi • WiFi hoppen • USB (For 3G/4G dongle)
Transmitting protocol	• MQTT
Installation way	• Screw
Waterproof/Dustproof	• IP54
Size	• 173*90*
Material	• ABS

ESL Specifications

Technical Specifications

Item	Description
Power	2 * CR2450 Battery
Work time	> 4 Years
Battery consumption	Average current < 25uA
Wireless distance	• Outdoor around 100 meters • Indoor around 30-50 meters (depends on environment)
Transmitting protocol	• BLE4.1/BLE5.0
Installation way	• 3M or Pylons
Total Size	• 93.4*45*11.9mm
Screen Size	• 2.9Inch
Material	• ABS

Outline

- **Introduction**

- **Solution Feature**

  - ✓ **Opening Platform Design**

  - ✓ **Self-organized deployed**

  - ✓ **OpenAPI for third part to integration**

  - ✓ **Security: Two-way authentication**

  - ✓ **Web portal Configuration UI**

1. Opening Platform Design

ESL Gateway not only an ESL manager. It's an **open platform to make your ESL and other BLE sensor easily connect to cloud.**
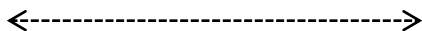
All the interface are opening for third part.
*   Support third part ESL/BLE device connect to the cloud via BLE GW;
*   Support third part MQTT server;
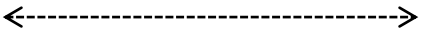*   Support your own firmware running on ESL Gateway;

ShenZhen KKM Co.ltd

Lissa market dep.

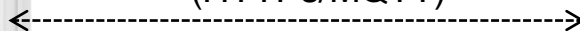Humidity sensor

iBeacon for navigation/promotion
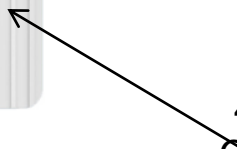
1. Opening authentication algorithm

2. Opening downdata/upload data interface

3. Opening JSON API (HTTPs/MQTT)
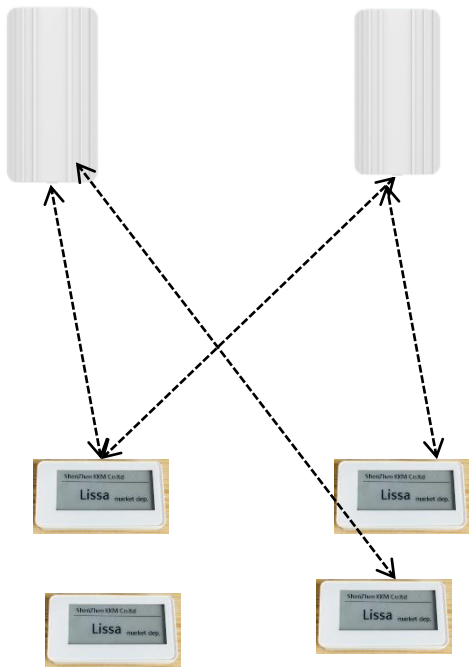
4. Opening source code on Openwrt, you can extend the service/message on BLE GW

2. Self-organized deployed

- Once you deployed the ESL(>1000pcs) and Gateway, the relationship between ESL and Gateway will **automatic setup within 3~5 minute**.

- Gateway supports **redundant configuration**, once one Gateway down, other Gateway(with in same coverage) will automatic take over ESL management within 10 seconds.

- Gateway support **pool load balance deployed**, the Customer Inventory System can select an light load Gateway to update price about ESL.
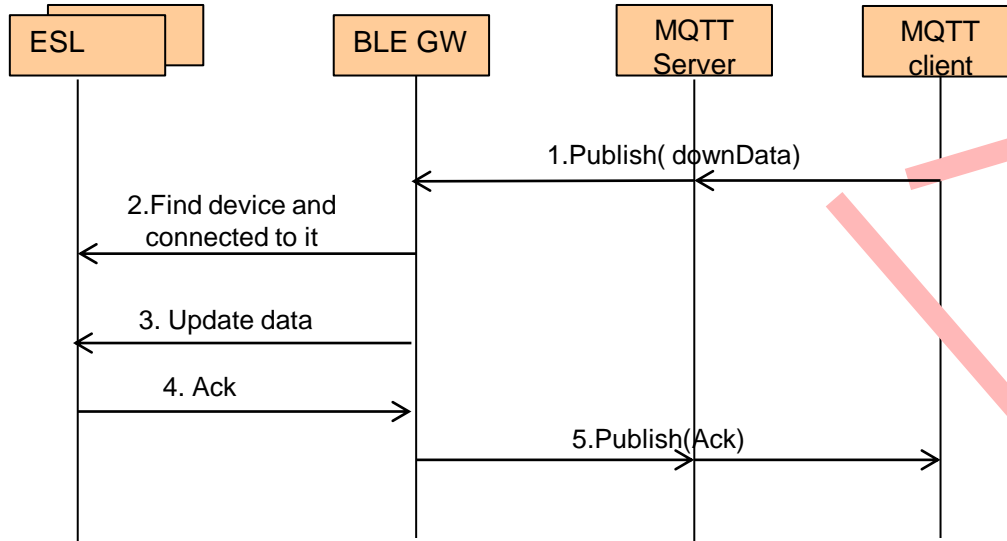
3. Open API Example: ESL monitor

Beacon
BLE GW
HTTPs/MQTT server

Status report

Status report

MQTT/HTTPs
(JSON: advData)
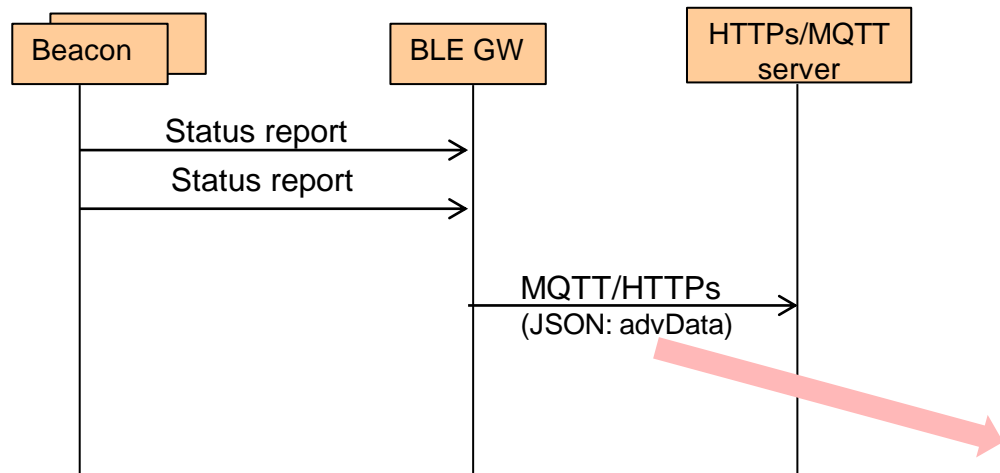
advData packet example

{
  "msg": "advData",
  "gmac": "66E865E28547",
  "obj": [
    {
      "dmac": "B45D0196FFFF",
      "rssi": "-53",
      "data1": "0201060302E0FF03095446"
    },
    {
      "dmac": "8D0389AB78CC",
      "rssi": "-92",
      "data1": "020106030202180AFF4B4D028D0389AB78CC"
    },
  ],
  "seq": 5091
}

3. Open API Example: ESL update

Download HEX data to beacon

```
{
  "msg": "dData",
  "mac": "e44892f98100",
  "seq": 10444,
  "auth1": "00000000",
  "dType": "hex",
  "data": "01000005C61280"
}
```

ESL   BLE GW   MQTT Server   MQTT client

1.Publish(downData)

2.Find device and connected to it

3. Update data

4. Ack

5.Publish(Ack)

modify the ESL status report period to 1000)

```
{
  "msg": "dData",
  "mac": "e44892f98100",
  "seq": 10444,
  "auth1": "password1",
  "dType": "json",
  "data": {
    "advPeriod": 1000,
    "txPwr": 5
  }
}
```

Support JSON data and hex data type to beacon

3. Open source code/SDK on Github

1. Gateway integration guidance and gateway installation and configuration guidance.
https://github.com/kkmhogen/ESLIntroduction.git

2. The client software Demo supports the connection of the MQTT server address and is based on the Java1.7 version.
https://github.com/kkmhogen/KGatewayClientDemo.git

3. Convert the bmp picture formatted image to the demo of the MQTT JSON message, based on the C language implementation, developed using Visual Studio 2012.
https://github.com/kkmhogen/Bmp2EslJson.git

We will continue to update the resources to make your develop more easily.

4. Two-way authentication ensures device security

To ensure that the devices and gateways are not illegally connected. BLE Gateway and device using Bi-directional MD5 with random authentication.

1.Verify device valid(MD5)

2. Verify BLE GW valid(MD5)

3. Tx/Rx data
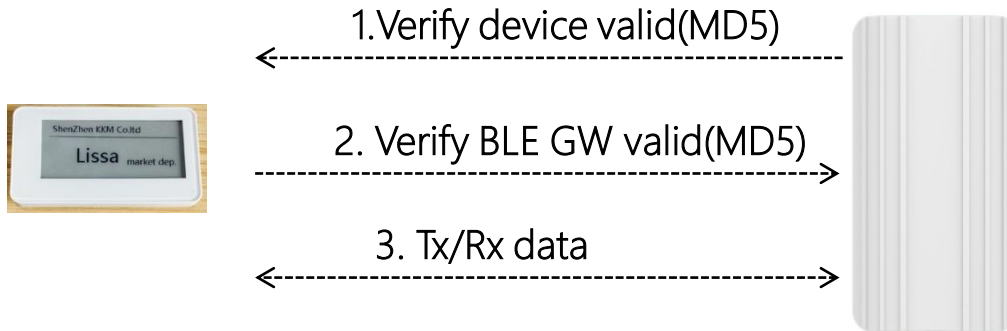
5. Web portal Configuration

The Gateway Support Web portal configuration, for example: chrome explore

• No need to install app

Thanks

www.kkmcn.com