

# KGateway(for KBeacon) Configuration Manual

## Revision History

Version	Date	Change Description	Author
V1.0	2017/7/1	initial	Ning
V1.1	2018/8/20	Add hidden Wi-Fi function	Hogen
V1.2	2018/12/1	Modify network architecture picture	Hogen
V1.3	2019/12/1	Add the Cellular info	Hogen
V1.31	2020/03/1	Add MQTT configuration example	Hogen

# Catalogue

1. Purpose.....	3
2. Introduction.....	3
3. External interface .....	4
3.1 Network interface.....	4
3.2 Power supply interface.....	4
3.3 LED indicator.....	4
3.4 Insert LTE USB dongle SIM card (If needed) .....	5
4. Configuration .....	5
4.1 Connect to KGateway .....	6
4.2 Network interface.....	7
4.2.1 Configure WAN interface .....	7
4.2.2 View the WAN interface status.....	10
4.2.1 Configure KGateway Wi-Fi AP.....	11
4.3 Service configuration .....	12
4.3.1 Advertisement Upload Parameters.....	12
4.3.2 MQTT without SSL configuration.....	14
4.3.3 MQTT with SSL configuration .....	15
4.3.4 HTTP configuration .....	18
4.4 Modifying the Web Portal Login Password.....	19
5. Quickly verify KGateway API.....	19
5.1 How to verify HTTP API.....	20
5.2 How to verify MQTT API.....	20
6. Trouble shooting .....	20
6.1.1 The KGateway flash red LED .....	20
6.1.2 The Cellular signal was very pool.....	25
7. Appendix1 Advertisement Upload Filter for different scenario .....	26
7.1 Scenario 1: Only upload nearby beacon advertisement .....	26
7.2 Scenario 2: Reduce advertisement message to clouds .....	27
7.3 Scenario 3: Only upload specific MAC address to clouds.....	28
7.4 Scenario 4: Only upload iBeacon advertisement .....	28
7.5 Scenario 5: Only upload Eddystone advertisement.....	29
7.6 Scenario 6: Filter advertisement packet by service ID.....	30
7.7 Scenario 7: Beacon Location .....	30
8. Appendix2 Setup your own MQTT Server.....	31

# 1. Purpose

This document describes the basic functions and physical interfaces of the KGateway, which are mainly used to guide users to install and configure.

# 2. Introduction

The KGateway is made of PC material and is waterproof and dustproof. It supports outdoor installation and can be installed by wall mounting.

After the KGateway power on, it will periodically scan the KBeacon advertisement packet then report the data through Wi-Fi or Ethernet. Also it can accept data command from the cloud and forward the data to KBeacon, such as updating the KBeacon configuration. The KGateway uses open MQTT + JSON API interface for third-party integration.



## Specification

Power	POE or DC 5V
Scanning ability	> 240 beacon per 1 second (nRF52840) > 120 beacon per 1 second (TI CC2640R2F)
Wireless distance	BLE5.0: > 200 meters BLE4.0/4.1/4.2 > 100 meters (depends on environment)
BLE	MAX TX power: 8dBm for nRF52840 5dBm for TI CC2640R2F Receiver Sensitivity: -96 dBm Protocol: BLE4.0/4.1/4.2/5.0 Antenna: omnidirectional Antenna VSWR: < 1.3
Transmitting way	• ETH RJ45

	<ul style="list-style-type: none"> <li>• Wi-Fi</li> <li>• Wi-Fi hopping</li> <li>• 3G/4G USB Dongle(optional)</li> </ul>
API protocol	<ul style="list-style-type: none"> <li>• HTTPS</li> <li>• MQTT</li> </ul>
Installation way	Screw
Waterproof/Dustproof	IP54
Size	173*90*40mm
Material	ABS

## 3. External interface

### 3.1 Network interface

The KGateway supports following ways to connect to the internet:

1. through Wi-Fi;
2. through the Ethernet cable;
3. Though Cellular network (optional, only available when the LTE usb dongle was insert);
4. The KGateway supports Wi-Fi Hopping, which means one KGateway can connect to internet by another KGateway.

### 3.2 Power supply interface

There are two interfaces for power supply: macro USB interface and Ethernet POE port;

- POE power supply, directly through the Ethernet cable interface, using POE(802.3af) to supply power.
- Macro USB power supply, powered by the 5V/1A DC.



**Warning: The KGateway can only use one of the two power supplies at the same time. Please don't insert two power supplies at the same time, otherwise KGateway may be damaged.**

### 3.3 LED indicator

The gateway has 2 LED indicators. The specific meanings are as follows:

### 1. Red indicator light:

If the red led flash, it means the gateway connect to the cloud fails.

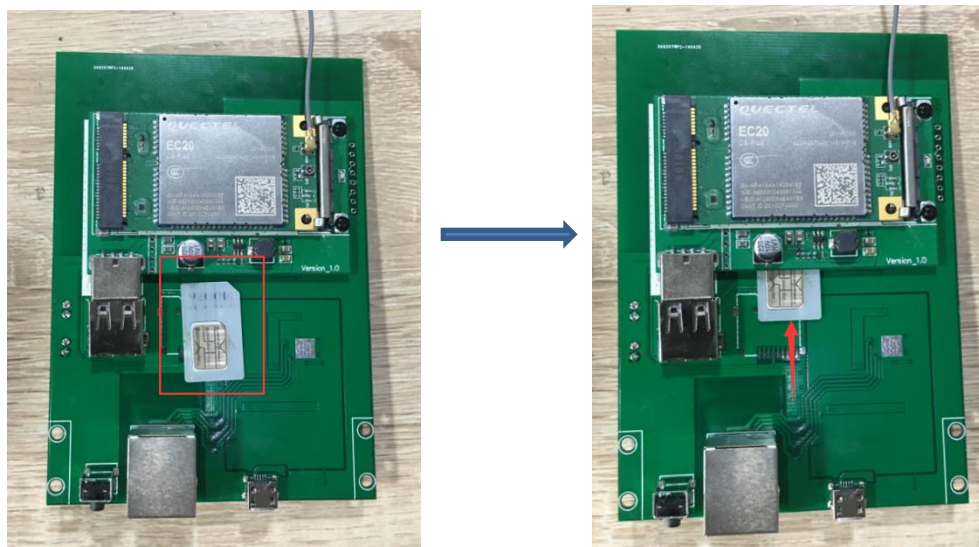
### 2. Green indicator light:

- 2-seconds or less frequency flash: indicates that the KGateway is successfully connected to the cloud and report KBeacon advertisement packet success.
- 10-second frequency flash: indicating that the KGateway connect to the cloud success, but it didn't find any KBeacon devices.

## 3.4 Insert LTE USB dongle SIM card (If needed)

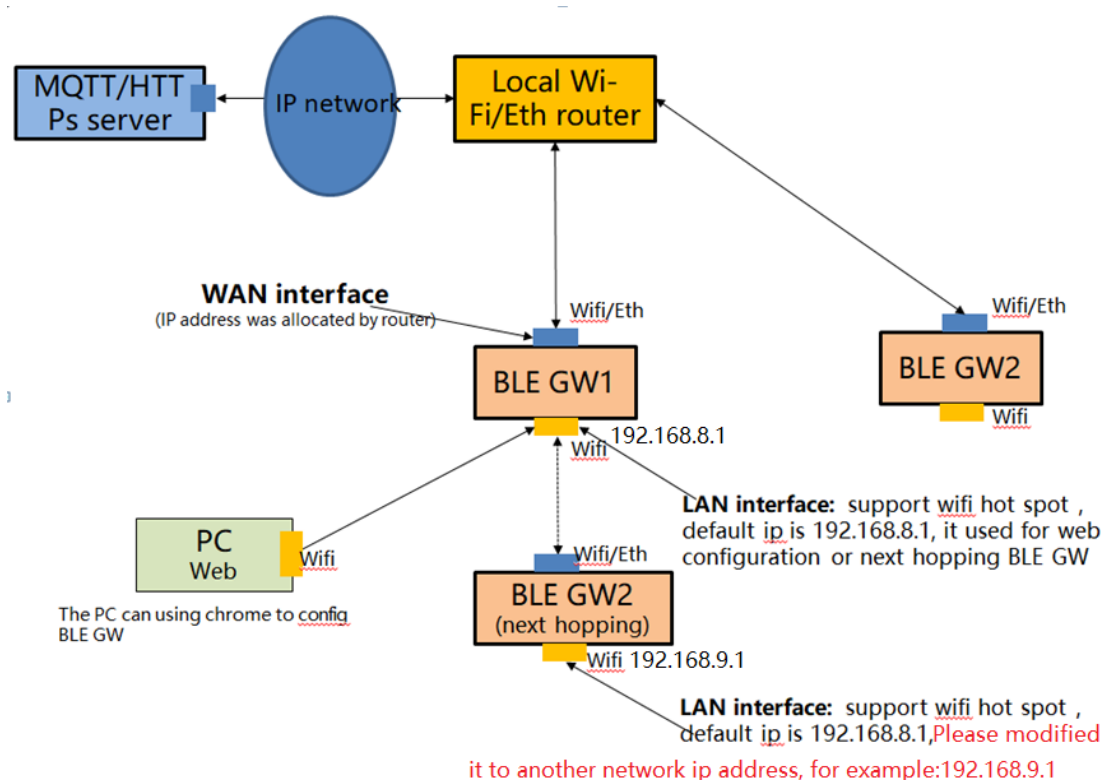
Please note: This procedure is only for KGateway with USB dongle and LTE module installed.

Disassemble the gateway and insert the SIM card at the position indicated in the picture below.



## 4. Configuration

The KGateway is configured in web portal mode. You can use an web browser to configure it. Chrome browser is recommended to use for the configuration.



As shown above, each base station has two interfaces with different IP addresses. One of these IP addresses (referred to as the WAN port) is used to connect to the internet network (MQTT server), which has a series of firewall rule protection. The other IP address (referred to as LAN port, also known as the intranet interface) is used for Wi-Fi hotspot broadcasting.

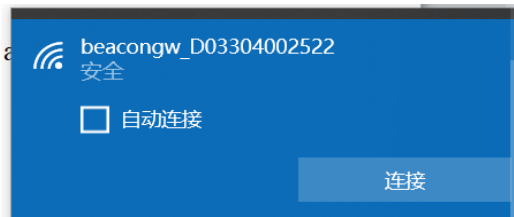
**WAN Port:** This interface supports Wi-Fi and ETH (network wire). KGateway can connect to routers via Wi-Fi or network wire, where IP addresses are assigned by routers. The base station is connected to the MQTT/HTTPs server through this interface, so you need to ensure that the network between this interface and the MQTT server is interconnected. WAN address IP address configuration see "4.3.2 configuration WAN port network connection".

**LAN port:** This interface only supports Wi-Fi. The default IP address of this interface is 192.168.8.1, PC can connect to this interface through Wi-Fi, or the next hopping KGateway can connect to the internet by Upper level.

If you need to configure the gateway, you can only configure it through Wi-Fi, and you cannot configure it through the network line (for security reasons, the network line interface only supports the WAN interface).

## 4.1 Connect to KGateway

- After power on, the KGateway will automatically broadcast Wi-Fi signal, and the default Wi-Fi name is "beacongw\_mac address"



- The default Wi-Fi connection password is “12345678”
- The default KGateway configuration IP address is 192.168.8.1
- Login in to the gateway by typing <http://192.168.8.1> in the browser.
- 5. Enter the user name: ‘admin’ and password: ‘admin’

## Admin

### Authorization Required

Please enter your username and password.

Username

admin

Password

Login

Reset

Powered by [LuCI for 15.05 branch \(git-18.097.75958-5537d68\)](#) / [OpenWrt Chaos Calmer 15.05.1 r49396](#)

## 4.2 Network interface

### 4.2.1 Configure WAN interface

Tap on **Network** to go to the network configuration page. You can choose to connect to the internet network using Wi-Fi /Ethernet/Cellular connection.

#### 4.2.1.1 Connect to internet by Ethernet

Admin	Status	Network	Service	Others	Logout
-------	--------	---------	---------	--------	--------

---

Wan Mode	Ethernet
----------	----------

---

Mode	dhcp
------	------

---

Apply

---

Show/Hide Wifi Name	<input checked="" type="checkbox"/>
---------------------	-------------------------------------

---

Apply

The IP address can be assigned in DHCP or static configuration.


#### 4.2.1.2 Connect to internet by Wi-Fi

Admin	Status	Network	Service	Others	Logout
-------	--------	---------	---------	--------	--------

---

Wan Mode	Wlan
----------	------


---

Nearby WLAN		
-------------	--	---

---

Name	
------	--

---

Password		
----------	--	---

---

Mode	dhcp
------	------

---

Apply

---

Show/Hide Wifi Name	<input checked="" type="checkbox"/>
---------------------	-------------------------------------

---

Apply



The screenshot shows the 'Nearby WLAN' dropdown menu with the following list of networks:

WiFi name	WiFi signal
KZLW	65
mabangbiaoju2	42
XCD	100
WINSKOM	68
e0:24:81:67:ef:3d	99
dc:16:b2:7d:6e:ad	76
e0:24:81:67:ef:39	39
ChinaNet-zi3x	100
1e:bc:5a:90:92:c0	91
1e:bc:5a:8f:d3:c0	100
WINSKOM	42
KKMdingding	91
KKMdingding	86
ChinaNet-YKhM	39
ChinaNet-hgpu	24
kkm_work_24G	96
kkm_guest	99
ChinaNet-uVkk	37
dc:90:88:32:33:d5	70
HMHX	70

Click on “Nearby WLAN”, then you can see the Wi-Fi name and the signal strength (0~100). 100 means the signal is very strong.

#### Connect to an hidden Wi-Fi AP:

You can see some Wi-Fi name like mac address; it means that Wi-Fi was hidden. You should input the Wi-Fi name and password if you want to connect to it.

The screenshot shows the configuration interface with the following fields:

- Wan Mode: Wlan
- Nearby WLAN: 1e:bc:5a:8f:d3:c0 | 100
- Name: (empty)
- Password: (empty)
- Mode: dhcp
- Apply button



If your Wi-Fi AP name does not appear in the nearby WLAN list, please try to reboot the KGateway. (In others page, you can reboot the device)

### 4.2.1.3 Connect to internet by Cellular

Admin
Status
Network
Service
Others
Logout

---

Wan Mode

Cellular

---

Mode

dhcp

---

Apply

---

Show/Hide Wifi Name
☒

---

Apply



Please make sure SIM card was insert into the KGateway.

## 4.2.2 View the WAN interface status

### 4.2.2.1 Ethernet status

AP MAC	D0:33:06:00:20:02
WAN MAC	D0:10:06:00:20:03
WAN Type	Ethernet
WAN IP	192.168.3.62
Router IP	192.168.3.1
DNS IP	192.168.3.1
UP Time(Sec)	2085
RX Bytes	35703856
TX Bytes	5856232
Cellular Info	N/A

The AP MAC was also printed in the KGateway shell.

WAN IP: The gateway IP address in WAN interface.

Router IP: Your router IP address.

UP time: How long the network interface was active.

RX Bytes: received data from WAN interface.

TX Bytes: sent data to WAN interface.

### 4.2.2.2 Cellular status

kkm	Status	Network	Service	Others	Logout
AP MAC	D0:33:04:01:59:12				
WAN MAC	D0:33:04:01:59:13				
WAN Type	Cellular				
WAN IP	10.24.169.207				
Router IP	10.24.169.208				
DNS IP	202.96.134.33; 202.96.128.166				
UP Time(Sec)	329261				
RX Bytes	1364937972				
TX Bytes	889406912				
Cellular Info	Type:Ite; RSRP:-99; SNR:72				




If the KGateway connect to internet by cellular, the cellular signal will show in this page.

- Very good signal: RSRP>-85dBm;
- Good signal: RSRP=-85~-95dBm;
- Middle signal: RSRP=-95~-105dBm;
- Pool signal: RSRP=-105~-115dBm;
- Very pool signal: RSRP<-115dB;

### 4.2.1 Configure KGateway Wi-Fi AP

Check status about KGateway, you can change the AP LAN IP address.

#### AP CONFIGURATION

AP SSID	<input type="text" value="beacongw_D03304002522"/>
AP LAN IP	<input type="text" value="192.168.8.1"/>
AP Password	<input type="password" value="....."/> 
<input type="button" value="Apply"/> <input type="button" value="refresh"/>	

## 4.3 Service configuration

Click Services to go to the service configuration page, where each field is defined as follows:

### 4.3.1 Advertisement Upload Parameters

Click Services to go to the service configuration page, where each field is defined as follows:

Upload period	<input type="text" value="2"/>
Redundant upload period	<input type="text" value="5"/>
Max Packet Size	<input type="text" value="60"/>
RSSI filter	<input type="text" value="-100"/>
ServiceID filter	<input type="text" value="0X0"/>
Ble Mac filter	<input type="text" value="Regular expression; e.g. ^20DD ^2"/>
Raw data filter	<input type="text" value="Regular expression; e.g. ^0201"/>
Upload iBeacon	<input type="text" value="YES"/>
Upload Eddystone	<input type="text" value="YES"/>
Upload KSensor	<input type="text" value="YES"/>
Upload Unknown	<input type="text" value="NO"/>
Include RSSI only	<input type="text" value="NO"/>
Service Access	<input type="text" value="MQTT"/>

**1. Upload period:** KGateway uses this parameter to control upload period of modified advertisement data about KBeacon to Cloud.

It needs at least **Upload period** seconds for KGateway to send advertisement to cloud when KBeacon advertisement data change.

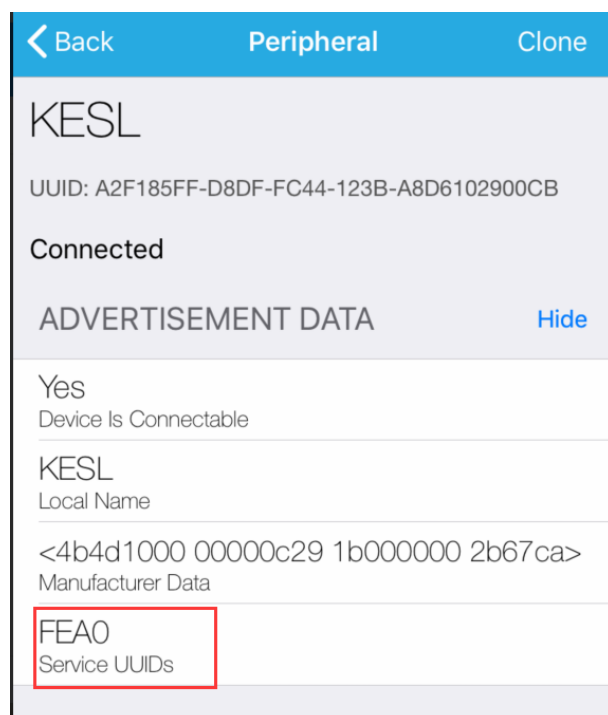
**2. Redundant Upload period:** KGateway uses this parameter to control upload period of unchanged advertisement data of KBeacon to cloud. The cloud uses this parameter to keep alive for all KBeacon.

**3. Max Packet Size:** This parameter is used to control max packet size when upload KBeacon advertisement data to KBeacon.

**4. Rssi filter:** If this parameter is set, the KGateway will only report the advertisement packet which signal is > **Min Rssi** value.

**5. Ble Services filter:** If this parameter is set, the KGateway will only report the advertisement packet which includes the setting BLE services ID.

*Following example is using Lightblue app on IOS to view the device service UUIDs, then you can set the services filter to 0xFEAO.*



**6. BLE mac filter:** KGateway can use this parameter to filter KBeacon mac address. This parameter uses Regular express.

For example, if Ble Mac filter value set to ^20DD, then following KBeacon advertisement packet will report to cloud.

- KBeacon1: ble mac = 0x20DD01000002 : report to cloud
- KBeacon2: ble mac = 0xA133DD010002 : not report to cloud
- KBeacon3: ble mac = 0xA10005033DD2 :not report to cloud

**7. Raw data filter:** KGateway can use this parameter to filter KBeacon advertisement packet. This parameter uses Regular express.

**8. Upload iBeacon:** ‘Yes’ means KGateway will report iBeacon protocol advertisement.

**9. Upload Eddystone:** ‘Yes’ means KGateway will report Eddystone(URL/TLM/UID) protocol advertisement.


**10. Upload KSensor:** ‘Yes’ means KGateway will report KKM KSensor protocol

advertisement.

**11. Upload Unknown:** ‘Yes’ means KGateway will report unknown advertisement that using hex string.

**12. Include RSSI only:** ‘Yes’ means KGateway will only report RSSI and not include advertisement content. This option will reduce the advertisement packet size.

### 4.3.2 MQTT without SSL configuration

Service Access	MQTT ▼
Url	api.ieasygroup.com:61613
Client ID	kb_client_D03304002122
Qos	0
User Name	kkmtest
Password	..... 
Publish Topic	kbeacon/publish/D03304002122
Pubaction Topic	kbeacon/pubaction/D03304002122
Subaction Topic	kbeacon/subaction/D03304002122
<input type="button" value="Apply"/>	


1. **Service Access:** select MQTT and the KGateway will using MQTT protocol to connect to cloud server.
2. **Url:** the MQTT cloud address.
  - TCP:// select TCP for connection.
  - MQTT URL address and MQTT port. The default port is 61613.
3. **Client ID:** Mqtt client id
4. **Qos:** MQTT qos value for publish action and subscribe action topic. The publish Topic

Qos is fixed to 0.

5. **Username:** mqtt client user name
6. **User Password:** mqtt client password
7. **Publish Topic:** The status of the status release message, the status of the gateway scanning to each label, published through this topic.
8. **Publish Action:** The response message of the gateway to the MQTT server, such as pictures and new response messages, is published through this topic.
9. **Subscribe Action:** The gateway store will subscribe to the request from the MQTT server to listen to this topic. Such as pictures with new request messages.

Other MQTT parameters are basic MQTT information, which will not be detailed here.

### 4.3.3 MQTT with SSL configuration

Url	api.ieasygroup.com:8883
Client ID	kb_client_D03306002002
Qos	0
User Name	kkmtest
Password	..... 
CA	-----BEGIN CERTIFICATE----- MIIDPzCCAo+gAwIBAgIJAI0op
Client Certificate	-----BEGIN CERTIFICATE----- MIIERjCCAy6gAwIBAgIJAN1Jjii
Certificate Key	-----BEGIN RSA PRIVATE KEY----- -----

1. **Service Access:** select MQTT and the KGateway will using MQTT protocol to connect to cloud server.
2. **Url:** the MQTT cloud address.
  - **SSL://** : select SSL for TLS connection.

- MQTT URL address and MQTT port. Our test MQTT server SSL port is 8883.
- 3. **Client ID:** Mqtt client id
- 4. **Qos:** MQTT qos value for publish action and subscribe action topic. The publish Topic Qos is fixed to 0.
- 5. **Username:** mqtt client user name
- 6. **User Password:** mqtt client password

**SSL Parameters,** the KGateway support self-signed certificates MQTT access.

7. **CA:** the CA file about the MQTT server

Default CA file about MQTT test server, please copy follow text to the CA field.

```
-----BEGIN CERTIFICATE-----
MIIDPzCCAo+gAwIBAgIJAI0p2GKzfvrMA0GCSqGSIb3DQEBAQUAMGoxFzAVBgNV
BAMMDkFuIE1RVFQgYnJva2VyMRYwFAYDVQQKDA1Pd25UcmFja3Mub3JnMRQwEgYD
VQQLDAtnZW5lcmF0ZS1DQTEhMB8GCSqGSIb3DQEJARYSbm9ib2R5QGv4YW1wbGUu
bmV0MB4XDTE5MDYyNzA5NDc1OFoXDTMyMDYyMzA5NDc1OFowajEXMBUGA1UEAwO
QW4gTVFUVCBicm9rZXIxZjAUBGNVBAoMDU93blRyYWNrcy5vcmcxFDASBgNVBAsM
C2dlbmVyYXRILUNBMSEwHwYJKoZIhvcNAQkBFhJub2JvZlIAZXhhbXBsZS5uZXQw
ggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCdpXqe1CtVzexiQYOhJ8l/
KYXcervHANiyBESzeBzA37kxK4hhAc7EuM8t0fuolOqHbRApc29sFXwC/mhz+xa72
FqVyAKeYUIREr808mcjyInvaaZJH9IgjAiAzONT38eNbvTbH9MHV2fD1SEZJozqH
mvW6BCeEjuPpvzljGva1tO9tXegheLPEM+J1bnm7i5uPtN26BgZxxAxcuoxFQSST
g8bNcw656+fPJf5YkLU8BKAIX0ezeZi/Y0olTU+UoZZ97v11onZ4QHc/YVB7KWX
ecGRnxXRee4sbXYIJuZ8ok/4k7LyFtKy54GhxojV9vsQj+AHmGN/H3hMzsjgOERL
AgMBAAGjUDBOMB0GA1UdDgQWBBSwp17hqpq8Ny+rEmpX2rfgz4AFrTafBgNVHSME
GDAWBgBswp17hqpq8Ny+rEmpX2rfgz4AFrTAMBgNVHRMEBTADAQH/MA0GCSqGSIb3
DQEBAQUAA4IBAQBE4mzg2XtTJCNP4rdi8Ti8RKEhB83R6PcUe8VOmF3oRmmE3uL+
IzgGJnXqNC6tXwmro+QyLQKUhdTMebsfxa9/WOQMWF2JQzZPX8tYff8QhBWYf1Y
z+pX8+5NWdwLryckmWPLeg9nNh9K80wHrgmqVm1efJZGGr3ph+72PTJUp8F0QKD4
IJuNdtw5juU07ji/WxM7KMF+OBBzNgXYzMGvN7nVdpIyM7fPKAFr4ZZal8mo9Qs+
wwQAJnCM2VhCMmEQJqwtYrA34bVxDoyGugd6O0iusrvoeijYZOXBDzrlvXCb1
6YcHk31LuRiKMymb6Llz8jHOz9pw9in6Gkru
-----END CERTIFICATE-----
```

8. **Client Certificate:** MQTT Client certificate file.

Default client certificate file about MQTT test server, please copy follow text to the Client Certificate field.

```
-----BEGIN CERTIFICATE-----
MIIERjCCAY6gAwIBAgIJAN1JjiHxuQFMA0GCSqGSIb3DQEBAQUAMGoxFzAVBgNV
BAMMDkFuIE1RVFQgYnJva2VyMRYwFAYDVQQKDA1Pd25UcmFja3Mub3JnMRQwEgYD
VQQLDAtnZW5lcmF0ZS1DQTEhMB8GCSqGSIb3DQEJARYSbm9ib2R5QGv4YW1wbGUu
```



```

bmV0MB4XDTE5MDYyNzA5NTIwN1oXDTMyMDYyMzA5NTIwN1owEzERMA8GA1UEAwI
bXljbGllbnQwgEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDLLzrOzsPK
vHKRjWuJuGpKe9pEk2e3Z0uD2XDGIpIAetDrTiW7bvmw8zWktT16VTWS7DcPFniQ
u9UKGmHILkIsIE4VOv2p6AoZOttDnsEcFzWqyeA2pVRv/G9M/nSLMUBA6XeQq+os
8HdLnx1uxL3auc9lj3SBFePRhhRchejEmNYdbTRWXWfNjgoqEnU4jR9+Fc+EPXXZ
0y+O0/9oAALooUI9GLTdMxSuw+Xeq5ZHmHGDTSYFg0wrigB3jeM/omrj2nEae5zP
yh0Dqxkbzz4qV+2yglh3eOmFtH2YC5ZKur8y5779H7uUiu+M5fZLk1tt0posMsO
hvebbBgMbhJjAgMBAAGjggFEMIIBQDAMBgNVHRMBAf8EAjAAMAKGA1UdEQQCMAAw
EQYJYIZIAyB4QgEBBAQDAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMCBggrBgEFBQcD
BDALBgNVHQ8EBAMCA6gwKAYJYIZIAyB4QgENBBsWGUNsaWVudCBCcm9rZXIqQ2Vy
dGlmaWNhdGUwHQYDVR0OBByEFEE0MxRznHh8gZcyT720Jc/wNQKDMIGcBgNVHSME
gZQwgZGAFLCnXuGqmrw3L6sSalfat+DPgAWtoW6kbDBqMRcwFQYDVQDDA5BbiBN
UVRUIGJyb2tlcjEWMBQGA1UECgwNT3duVHJhY2tLm9yZzEUMBIGA1UECwwLZ2Vu
ZXJhdGUtQ0ExITAFBgkqhkiG9w0BCQEWEm5vYm9keUBleGFtcGxlLm5ldIIIjAI0o
p2GKzfvrMA0GCSqGSIb3DQEBDQUAA4IBAQc51d61OUeZGgj6GCmoiOsSNSHkf8qe
mT//Sb67D51Je234H5Qjo57Fp270/P9aBZJeUOyb6vLtQdlrAPCLpkjatKjjUZY
8MqKohDDKATEo7J8+VAUU+1RHH1rL7qTke7bqQA1D63X4uMCoqnENNvIrdx0aPs3
FbJamviEWEHE7Xe/9z/ttkD46iH6nK94FR4K1yFFq116VDXyj/8COA15lgxBqP9
i6lfDq3d6CrPxL8p1REAY/TigRD6jkKIr0xRbT8iMKMfgEknkmPvUSwB97Lom+Bg
exUIhP4E/g/JH3vUFbUAWSNwbesLIVux0iE3xLV3NQANXjbJL2ycqmny
-----END CERTIFICATE-----

```

## 9. Certificate Key: MQTT Client Key file:

Default client certificate key file about MQTT test server

-----BEGIN RSA PRIVATE KEY-----

```

MIIEowIBAAKCAQEAYy86zs7DyrxykScLibhqSnaRJNnt2dLg9lwxoqSAHrQ604l
u275sPM1pLU9eU1kuw3DxZ4kLvVChph5S5CLCBOFT9qegKGTTrbQ57BHBc1qsng
NqVUb/xvTP50izFAQOI3kKvqLPB3S58dbS92mPZY90gRXj0YYUXIXoxJjWHW00
Vl1nzY4KKHJ1OI0ffhXPhD112dMvjtp/aAAC6KFCPRi03TMUrsPl3quWR5hXg00m
BYNMK4oAd43jP6Jq49pxGnucz8odA6sZG88+KlftsoJYd3jphbR9mAuwSrq/Mue+
/R+7lIrvjOX2S5NbbbdKaLDLDob3m2wRpgYSYwIDAQABAoIBAHDilZ6uu5LYriI8
+ikS32Lweqzc5KO97POdDA4SkNIZYtdIvKftGSaZiTUwSK69xp0RRenJB0UKRmtt
ivQkhjkz3Y3oD2SleW7ZB1cDTbDIqFVyAXL62fW53zB8ss1emJ831uj2yo0bMHxA
M78F6aEq2LJJSgyrfj47avaYo2ENt370nUJVBWFzQtEq4lyvrHV561XC0PpNNOOc
hcBV3m9jycDzecsHt40X8aNk9Yu7rIbBvsLcUFpYFRpIU5PkdcMFXEBVNhvvA6Zn
jqilal8tKKc7CV4KupduXSxrHKoKyRdJsfvNoiBXVAcrO+mV4P1FsvWST478U5/p
E4CV0mECgYEA5r9o5UCWk+9XaMFXuB7w3cKCYV3TMs7wzyARaWCuVvOEKi28nZ59
DV7NJB80bJoFV0Y3IKDqLT0j8WdnkWOsiexOMPmMqD5nK7t4YjN7KzjYtsnOOVqE
/rYitWu48jOU2uHHJEYeXeoEX7do3KgEsgix16fmrq6+M8JAXJDvxECgYEA4Wud

```

```
G9SiSIPOh6xXEqBQZJj9EB8vsmHRwSLaLAaMXeYaViOHIA9ZA3JcwVEQIFCq4tGI
E1xg6G1S1kUJQj3/jjM15j7RldjxYh34KxA99hsSgTqzEA1vX9aLW8e5OGtE1WsE
NHKgRUJvL2qgA8Y07ErBmVft8ykn1byR1io64jMCgYBIkLVqMy+ngkNY3ES70G8r
LjpWaG08szujx4/yNcdd7OMJaKzCXM8y3zdgPrf710wKsDhWc2xxoaKmDDWG0XO3
RCsq1ay4iDdl12T6bPo1E4GDq19CRidA+wJbT28wK+yKESrRS4PNPvmoKEO/8M7Z
NkbHbTPvDzYxfNp2K/sGIQKBgQCOwpmIQ5zYV8UntWSbnj3rx9wU9rs9TKgLmef0
E92Xph+tx+iYQzQLc8T8hTsiTBQ9e5ejM1EIQhxBKQ2ZDyymwRX0kHopXXBZxEMj
+rBTCYgpoaQ/mVKcroIpiQ5rhbvdMV5DXQjVdCJ4NB3M5h0kPJ/pXn+LDNh2i2sM
6EfMcwKBgG72O5CFfxD50266IOz7MsUV0wIPN2kkhDB5oSUiJvc2i/piy+O0zCFE
TIEecsxxKWvLtoQBdlTgq9u15sFVSkKq5I2NQLjDhtCB5+alv3YzKpuv6lg0GIH2
/9T75FmQ0StYMJcWkOVmrfUQm7fNzFZv06CVIHkF3PaH823/p4Gs
-----END RSA PRIVATE KEY-----
```

10. **Publish Topic:** The status of the status release message, the status of the gateway scanning to each label, published through this topic.
11. **Publish Action:** The response message of the gateway to the MQTT server, such as pictures and new response messages, is published through this topic.
12. **Subscribe Action:** The gateway store will subscribe to the request from the MQTT server to listen to this topic. Such as pictures with new request messages.

### 4.3.4 HTTP configuration


Service Access	HTTP ▼
Authentication	None ▼
Url	<a href="http://api.ieasygroup.com:8091/post">http://api.ieasygroup.com:8091/post</a>

1. **Service Access:** select HTTP and the KGateway will using HTTP protocol to connect to cloud server.
2. **Authentication:** The KGateway support single password authentication for HTTP post.
3. **Url:** The cloud HTTP service.

We provide a test HTTP server and the URL address is:





<http://api.ieasygroup.com:8091/postdata>

Also the KGateway can support simple password that connect to clouds.

Authentication	<input type="text" value="BasicAuth"/>
Url	<input type="text" value="http://api.ieasygroup.com:8091/pos"/>
User Name	<input type="text" value="admin"/>
Password	<input type="password" value="....."/> 
<input type="button" value="Apply"/>	



## 4.4 Modifying the Web Portal Login Password

The default login password is “root” and users can change it to other password.

Admin		Status		Network		Service		Others	Logout
-------	---	--------	---	---------	---	---------	---	--------	--------

### Password Change

Password	<input type="password"/>	
Confirm	<input type="password"/>	
<input type="button" value="Apply"/>		

### Reboot

Reboot	<input type="button" value="Reboot"/>
--------	---------------------------------------

### Flash operations

## 5. Quickly verify KGateway API

In order for customer easily integrates our gateway, KKM provides test servers for HTTP

and MQTT.

## 5.1 How to verify HTTP API

1. The gateway is setting to MQTT server by default after factory. So please reference section<< 4.3.4 HTTP configuration>> to change the service's type.

2. KKM provides a test HTTP server, and the address is:

<http://api.ieasygroup.com:8091/postdata>

3. After KGateway connect to the HTTP service success, it will flash green LED and periodically post the KBeacon advertisement data to HTTP server.

4. You can view the reported data on HTTP server by follow address. You should replace the mac address to your KGateway.

<http://api.ieasygroup.com:8091/viewdata.jsp?mac=D03304001182>

api.ieasygroup.com:8091/viewdata.jsp?mac=D03304003262		
#Ble gate way data		
mac	data1	time
020100544EC0	{"rssi":-43,"dmac":"020100544EC0","time":"2020-01-08 00:44:07","type":16,"url":"026E74702E696D2F41514E4D5F357A397741","refPwr":4}	2020-01-08 08:44:08.0
CC1E24EA7DE0	{"rssi":-67,"majorID":1,"dmac":"CC1E24EA7DE0","minorID":1,"refpower":-59,"time":"2020-01-08 00:44:06","type":4,"uuid":"7777772E6B6B6D636E2E636F6D000001"}	2020-01-08 08:44:08.0

## 5.2 How to verify MQTT API

Download mqttfx software:

<http://www.jensd.de/apps/mqttfx/1.5.0/>

Please reference the document << KGateway API Specification.pdf>> for other details.

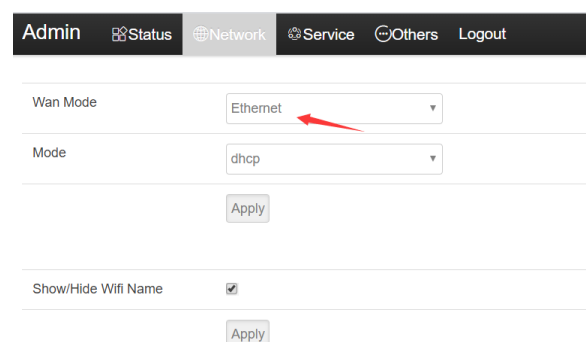
## 6. Trouble shooting

### 6.1.1 The KGateway flash red LED


If the KGateway connect to HTTPs/MQTT server success, it will flash green LED, otherwise it will flash red LED.

If the KGateway flash red led, please check the connection by following steps:

### 6.1.1.1 Step1: Check if the network connection is normal



Admin Status **Network** Service Others Logout

Wan Mode Ethernet 

Mode dhcp

Apply

Show/Hide Wifi Name ☒

Apply

Check if the network type is right.

≡ | 192.168.8.1/cgi-bin/luci/stok=e780db0884ca5c1ec7700d4b75eab50d/admin/hstatus

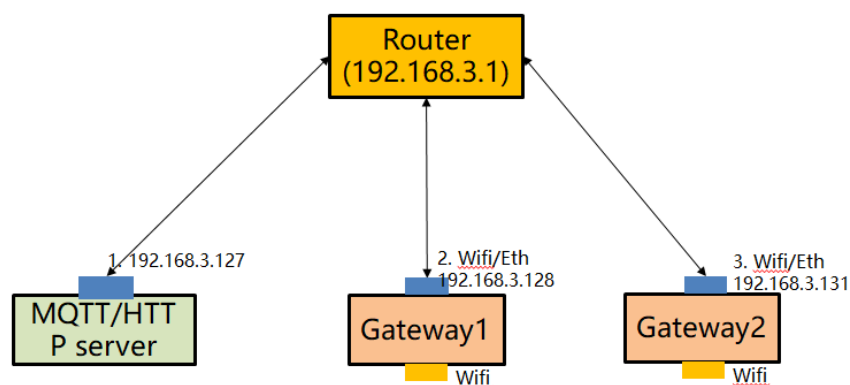
Admin Status Network Service Others Logout	
AP MAC	D0:33:04:00:64:52
WAN MAC	D0:10:04:00:64:53
WAN Type	Ethernet
WAN IP	192.168.3.162
Gateway IP	192.168.3.1
DNS IP	192.168.3.1
UP Time(Sec)	700
RX Bytes	5541423
TX Bytes	1259727
Cellular Info	N/A

#### Check the network status:

If Gateway connects network success, it will get the IP address and DNS IP address. Also the RX Bytes and TX Bytes will increase.

### 6.1.1.2 Step2: Check the connection between KGateway and cloud server

#### Scenario 1: The KGateway and Server are deployed in the same LAN



It is necessary to ensure that the base station and server are in the same network, that is, the assigned IP address is in the same network. For example, if the IP address of the MQTT server is 192.168.3.127, the base station 1 can be configured as the following address (192.168.3.128).

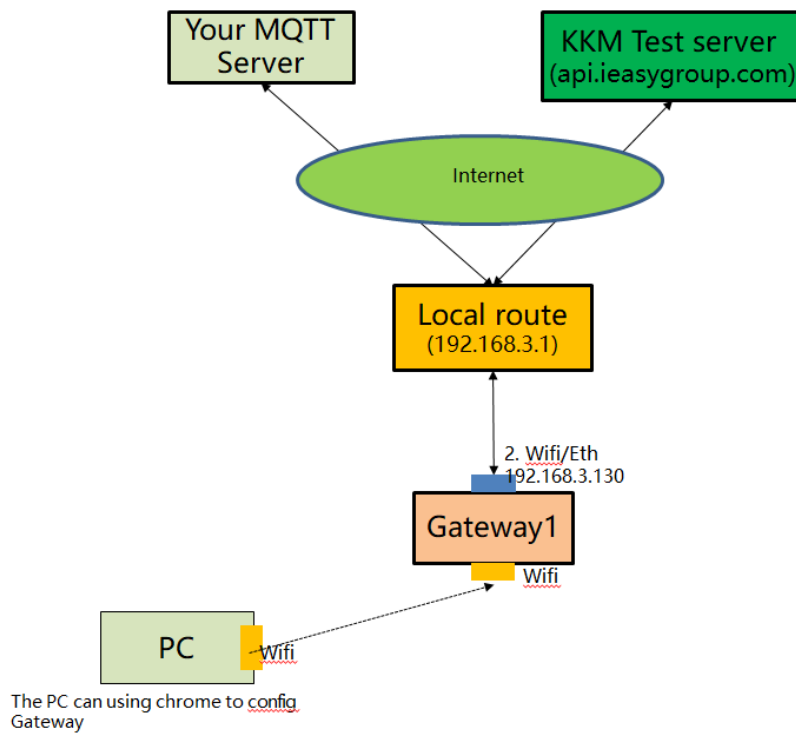
Admin	Status	Network	Service	Others	Logout
Wan Mode <input type="text" value="Ethernet"/>					
Mode <input type="text" value="static"/>					
IP Address <input type="text" value="192.168.3.128"/>					
Netmask <input type="text" value="255.255.255.0"/>					
Gateway Address <input type="text" value="192.168.3.1"/>					
Primary DNS Address <input type="text" value="192.168.3.1"/>					
Secondary DNS Address <input type="text" value=""/>					
<input type="button" value="Apply"/>					

Try to using PING command detection KGateway on MQTT/HTTP server. The PING command is used to detect whether the base station and the MQTT server network are connected. If the ping failed, please check whether the LAN is normal.

- Type: "Ping 192.168.3.128" on MQTT/HTTP server.

## Scenario two: the Servers are deployed in the cloud

Please try to use KKM test server to verify if KGateway connect to cloud success. We provide a cloud-based test server with the IP address of the MQTT server: api.ieasygroup.com; it support both MQTT and HTTPs. The KGateway has been set up as the KKM test server by default.

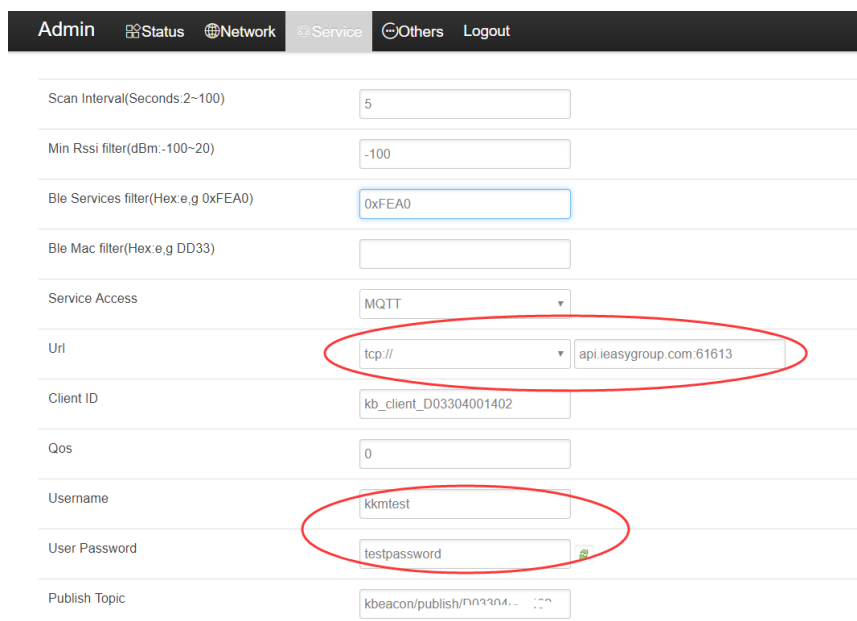


Assuming that the IP address of the router is 192.168.3.1, the IP address of the base station can be configured as 192.168.3.128.

Admin	Status	Network	Service	Others	Logout
Wan Mode	Ethernet				
Mode	static				
IP Address	192.168.3.128				
Netmask	255.255.255.0				
Gateway Address	192.168.3.1				
Primary DNS Address	192.168.3.1				
Secondary DNS Address					
Apply					

KKM MQTT server information:

- Address: api.ieasygroup.com:61613
- Test user name: kkmtest
- password: testpassword



Field	Value
Scan Interval(Seconds:2~100)	5
Min Rssi filter(dBm:-100~20)	-100
Ble Services filter(Hex:e.g 0xFEAE0)	0xFEAE0
Ble Mac filter(Hex:e.g DD33)	
Service Access	MQTT
Url	tcp://api.ieasygroup.com:61613
Client ID	kb_client_D03304001402
Qos	0
Username	kkmtest
User Password	testpassword
Publish Topic	kbeacon/publish/003304001402

Wait 30 seconds to 1 minute after saving the settings. If the base station flashes green light, the network connection between the base station and the cloud is normal. If it still flash red LED, the network connection between KGateway and the Cloud maybe failed.

### 6.1.1.3 Check if the HTTP/MQTT server running normal

#### Scenario 1: Using MQTT server

Please refer to section 7.2 Using third parts MQTT client to verify KGateway in <<KGateway API Introduction>> document.

If MQTT client connection fails, please check:

- 1) is there a firewall on the MQTT server to prevent other client connections? The default port of the MQTT server is 61613.
- 2) Whether the MQTT server is installed correctly.

#### Scenario 2: Using HTTPs server

You can use curl tools to verify if the connection and the key file is right.

Example:

```
curl --request POST --url 'http://api.ieasygroup.com:8091/postdata' --header 'content-type: application/json' --data '{ "msg": "advData", "gmac": "A1A2A3A4A5A9", "obj": [{"dmac": "AE9639C51701", "rssi": "-25", "data1": "020106"}, {"dmac": "7E4395AB78CC", "rssi": "-25", "data1": "020106030202180AFF4B4D027E4395AB78CC"}] }' --include
```

pause

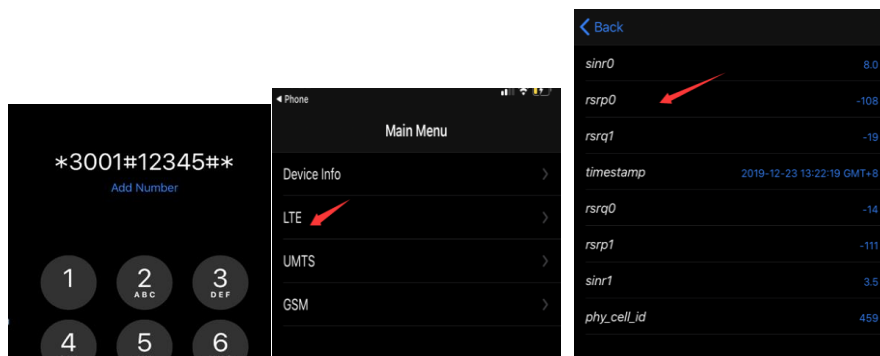


## 6.1.2 The Cellular signal was very pool

If cellular signal  $< -110\text{dBm}$ , it means the LTE signal was pool. Please check like follow:

Step1: Using iPhone to check your cellular provide signal.

Input `*3001#12345#*` on the dial UI, then tap on call.



Please check the rsrp0 signal.

You can goto step2 if the iPhone's signal was much bigger then KGateway showing. If it is almost same, please contact your cellular provider.

Step2: Open the shell box; please make sure that the antenna interface is firmly inserted.

*The antenna may be loosened when the case is opened and insert SIM card, so we check the antenna interface.*



## 7. Appendix1 Advertisement Upload Filter for different scenario

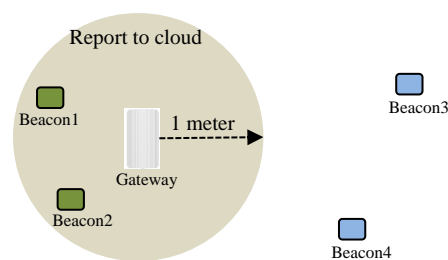
KGateway supports a variety of filtering conditions to meet the filtering requirements in different scenarios.

Upload period	<input type="text" value="2"/>
Redundant upload period	<input type="text" value="5"/>
Max Packet Size	<input type="text" value="60"/>
RSSI filter	<input type="text" value="-100"/>
ServiceID filter	<input type="text" value="0X0"/>
Ble Mac filter	<input type="text" value="Regular expression; e.g. ^20DD ^2"/>
Raw data filter	<input type="text" value="Regular expression; e.g. ^0201"/>
Upload iBeacon	<input type="text" value="YES"/>
Upload Eddystone	<input type="text" value="YES"/>
Upload KSensor	<input type="text" value="YES"/>
Upload Unknown	<input type="text" value="NO"/>
Include RSSI only	<input type="text" value="NO"/>
Service Access	<input type="text" value="MQTT"/>

### 7.1 Scenario 1: Only upload nearby beacon advertisement

Sometimes we want the KGateway only report the KBeacon that nearby KGateway.

For example, the Gateway is deployed on door, then we need the Gateway only report the beacons signal to clouds which is near the door.



Set the min RSSI filter to -59dBm.

Upload period	<input type="text" value="2"/>
Redundant upload period	<input type="text" value="5"/>
Max Packet Size	<input type="text" value="60"/>
RSSI filter	<input type="text" value="-59"/>

## 7.2 Scenario 2: Reduce advertisement message to clouds

Sometimes we may use third part MQTT hub to receive advertisement. Then we need to reduce the advertisement message number. Also some MQTT hub may limit the max MQTT message size.

For example:

- If the advertisement packet does not change, The Gateway use a long upload period interval to send the advertisement packet to clouds. We set to 60 seconds.
- If the advertisement packet changed, the Gateway sends the advertisement packet to cloud immediately. We set to 2 seconds.
- The max packet size is set to 60KB.

Upload period	<input type="text" value="2"/>
Redundant upload period	<input type="text" value="60"/>
Max Packet Size	<input type="text" value="60"/>

### 7.3 Scenario 3: Only upload specific MAC address to clouds

The KBeacon default mac address starts with DD33. And the BLE mac address is reversed; it means the BLE mac address end with 33DD.

We can set the BLE mac filter to 33DD\$ to filter KBeacon device. The Gateway only report KBeacon advertisement packet to clouds.

Upload period	<input type="text" value="2"/>
Redundant upload period	<input type="text" value="5"/>
Max Packet Size	<input type="text" value="60"/>
RSSI filter	<input type="text" value="-100"/>
ServiceID filter	<input type="text" value="0X0"/>
Ble Mac filter	<input type="text" value="33DD\$"/>

### 7.4 Scenario 4: Only upload iBeacon advertisement

Sometimes we want the KGateway only report the iBeacon advertisement packet to cloud, then we can set iBeacon to 'Yes' and others to 'NO'.

Upload iBeacon	<input type="text" value="YES"/>
Upload Eddystone	<input type="text" value="NO"/>
Upload KSensor	<input type="text" value="NO"/>
Upload Unknown	<input type="text" value="NO"/>
Include RSSI only	<input type="text" value="NO"/>

```
{  
  "msg": "advData",  
  "obj": [  

```

```

{
  "dmac": "51DC0EA4AE30",
  "refpower": -75,
  "uuid": "FB349B5F80000080001000003CFE0000",
  "majorID": "4115",
  "rssi": -80,
  "minorID": "077F",
  "type": 4,
  "time": "2019-09-02 09:47:42"
},
{
  "dmac": "231824EA7DE0",
  "refpower": -59,
  "uuid": "7777772E6B6B6D636E2E636F6D000001",
  "majorID": "0001",
  "rssi": -64,
  "minorID": "0001",
  "type": 4,
  "time": "2019-09-02 09:47:43"
}
],
"gmac": "D03304002122"
}

```

## 7.5 Scenario 5: Only upload Eddystone advertisement

Sometimes we want the KGateway only report the Eddystone advertisement packet to cloud, then we can set Eddystone to 'Yes' and others to 'NO'.

Upload iBeacon	NO
Upload Eddystone	YES
Upload KSensor	NO
Upload Unknown	NO
Include RSSI only	NO

```

{
  "msg": "advData",
  "obj": [
    {
      "dmac": "0A2024EA7DE0",
      "advCnt": 13586020,
      "vbatt": 3050,
      "secCnt": 13655980,
      "temp": 33,
      "time": "2019-09-02 09:51:11",
      "rssi": -63,
      "type": 8
    }
  ]
}

```

```

    },
    {
      "dmac": "7996010A33DD",
      "advCnt": 13848450,
      "vbatt": 3113,
      "secCnt": 13917330,
      "temp": 26,
      "time": "2019-09-02 09:51:12",
      "rssi": -75,
      "type": 8
    }
  ],
  "gmac": "D03304002122"
}

```

## 7.6 Scenario 6: Filter advertisement packet by service ID

The BLE advertisement packet can include Services ID. For example, the Eddystone beacon packet's services ID is 0xFEAA. If we set the service id, then the KGateway will only report Google Eddystone packet.

ServiceID filter

## 7.7 Scenario 7: Beacon Location

Sometimes the clouds only need to monitor beacon's RSSI for location. Then the Gateway only needs to scan device's RSSI and mac address. Because the Gateway will not scan advertisement packet data, so we cannot set the Upload iBeacon/Eddystone/KSensor to 'YES'.

In this scenario, the Gateway will only report the beacon's RSSI and mac address to clouds.

Upload iBeacon	<input type="text" value="NO"/>
Upload Eddystone	<input type="text" value="NO"/>
Upload KSensor	<input type="text" value="NO"/>
Upload Unknown	<input type="text" value="YES"/>
Include RSSI only	<input type="text" value="YES"/>

**advertisement uploaded:**

```

{
  "msg": "advData",
  "obj": [{

```

```

        "dmac": "3636000A33DD",
        "data1 ": "",
        "type": 32,
        "time": "2019-09-02 09:22:22",
        "rssi": -59
    }, {
        "dmac": "5055010A33DD",
        "data1 ": "",
        "type": 32,
        "time": "2019-09-02 09:22:23",
        "rssi": -44
    }, {
        "dmac": "7355010A33DD",
        "data1 ": "",
        "type": 32,
        "time": "2019-09-02 09:22:25",
        "rssi": -45
    }
  ],
  "gmac": "D03304002122"
}

```

## 8. Appendix2 Setup your own MQTT Server

There is some third-party MQTT server software. Following example uses mosquitto as an example which test in windows10 environment.

1. Download mosquitto:

<https://mosquitto.org/files/binary/>

2. We installed the software to C:\Program Files\mosquitto

3. Create the password file with username:

```
mosquitto_passwd -c pwfile2.example test
```

3. edit mosquitto.conf file,

add follow line in the file

```

max_connections -1
listener 61613
protocol mqtt
allow_anonymous false
password_file pwfile2.example

```

uncomment follow line:

```
log_timestamp true
log_timestamp_format %Y-%m-%dT%H:%M:%S
websockets_log_level 0
```

#### 4. Run mqtt server

```
mosquitto -c mosquitto.conf
```

#### 5. Verify mqtt server

a. Subscribet all topic: we assume the username is test and the password is abcabc.

```
mosquitto_sub -h localhost -p 61613 -t mqtt -u test -P abcabc
```

b. publish message to mqtt server:

```
mosquitto_pub -h localhost -p 61613 -t mqtt -m "hello world" -u test -P abcabc
```

NOTE: 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

Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.



This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

**RF Exposure Warning Statements:**

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment shall be installed and operated with minimum distance 20cm between the radiator & body.