**AEROSENSE DEVICE**

# I. Configuration devices through Aerosense-Wavve Tool

To configuration the device, you need to use tool in folder named “AeroSenseWavveTool 2.0.31.20240430\_Release”.

A screenshot of a computer

Description automatically generated

Power on the AeroSense Wavve device, Click Scan to scan for nearby Bluetooth devices. Select the Bluetooth with the name " AeroSense Wavve ". If the Bluetooth name is not found, press and hold the reset button on the back of the device for 3 seconds to reset the device (the blue LED on the device will blink), and then click Scan to search again. Click Connect to connect to the AeroSense Wavve device. If the connection is successful, the **Bluetooth State** will be Connected.

A screenshot of a computer screen

Description automatically generated

Enter the name and password of your Wi-Fi network and click Set. Enter the local IP address and **port** and click Set. To create a Server with Local IP by AerosenseWavve-SDK.

A screenshot of a computer

Description automatically generated

Check IP of server: Using “ipconfig” on PowerShell. After that, reset the power of the device.

A screenshot of a computer

Description automatically generated

# II. Install and Configuration Python environment

Download Python by visiting the official website: [Python.org](https://www.python.org), and downloading the latest stable version for your operating system (Windows, macOS, or Linux).

A screenshot of a computer

Description automatically generated

For Windows:

* Run the installer.
* Make sure to check the box **"Add Python to PATH"** during installation.

Install the libraries needed, use the following command:

|  |
| --- |
| pip install numpy  pip install paho-mqtt |

# III. Configuration AWS

Login to AWS Cloud ([AWS Console - Signup](https://signin.aws.amazon.com/signup?request_type=register)). After login successfully, choose ““AWS Management Console” and search “IoT Core”.

A screenshot of a computer

Description automatically generated

Choose “Manage” “All devices” “Things” “Create Things”.

A screenshot of a computer

Description automatically generated

Choose “Create single thing” and press “Next”.

A screenshot of a computer

Description automatically generated

Enter a name for the thing and click "Next".

A screenshot of a computer

Description automatically generated

Select “Auto-generate a new certificate” (or use “Use my certificate” if you already have one) and click "Next".

A screenshot of a certificate

Description automatically generated

Click “Create Policy”.

A screenshot of a computer

Description automatically generated

Enter a policy name and click “JSON”.

A screen shot of a computer

Description automatically generated

Add the following fields to the Policy document and click “Create”.

|  |
| --- |
| {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Allow",  "Action": "iot:Publish",  "Resource": "\*"  },  {  "Effect": "Allow",  "Action": "iot:Subscribe",  "Resource": "\*"  },  {  "Effect": "Allow",  "Action": "iot:Connect",  "Resource": "\*"  },  {  "Effect": "Allow",  "Action": "iot:Receive",  "Resource": "\*"  }  ]  } |

Return to the “Attach policies to certificate” page, select the policy you just created, and click “Create thing”.

A screenshot of a computer

Description automatically generated

In the pop-up window, click to download the device certificate, public key file, private key file, and root CA certificate (RSA 2048).

A screenshot of a computer

Description automatically generated

Modify the name of the certificate:

* Device certificate: certificate.pem.crt
* Private key file: private.pem.key
* Root CA certificate: root-CA.crt

# IV. Modify Parameters Inside Code

There are some parameters need to modify in code:

* AWS\_IOT\_ENDPOINT # Replace with your endpoint
* MQTT\_TOPIC\_PUBLISH # Replace with the topic you want to use
* root\_ca\_path # Path to root certificate
* private\_key\_path # Path to private key
* cert\_path # Path to certificate
* HOST #Replace your local IP

To get the “AWS\_IOT\_ENDPOINT”, open the AWS IoT and click Connect Connect One Device

A screenshot of a computer

Description automatically generated

**Note: The IP and PORT in code must match the configuration of the AeroSense Wavve Tool**.

# V. Run code

To run code, enter:

|  |
| --- |
| python main\_multi\_device\_aws.py |

To demonstrate in AWS IoT, click Test MQTT test client, enter your MQTT\_TOPIC\_PUBLISH and click “Subscribe”.

A screenshot of a subscribed

Description automatically generated

After subscribing, select the subscribed topic, and you will see the data reported by the Aerosense Device.

A screenshot of a computer

Description automatically generated