Sending IMU Data to AWS IoT Core and Setting Up a Rule to send it to AWS Lambda

1. **Overview:**

This document describes how the data acquired from the SenseHat Emulator in the Raspberry Pi is sent to the AWS IoT Core through the MQTT publish subscribe model.

1. **Pre-requisites:**

* The IoT device must be registered in the AWS IoT Core and the necessary certificates must be generated in order for the device to communicate with the cloud. The following link is the official AWS documentation for this purpose:

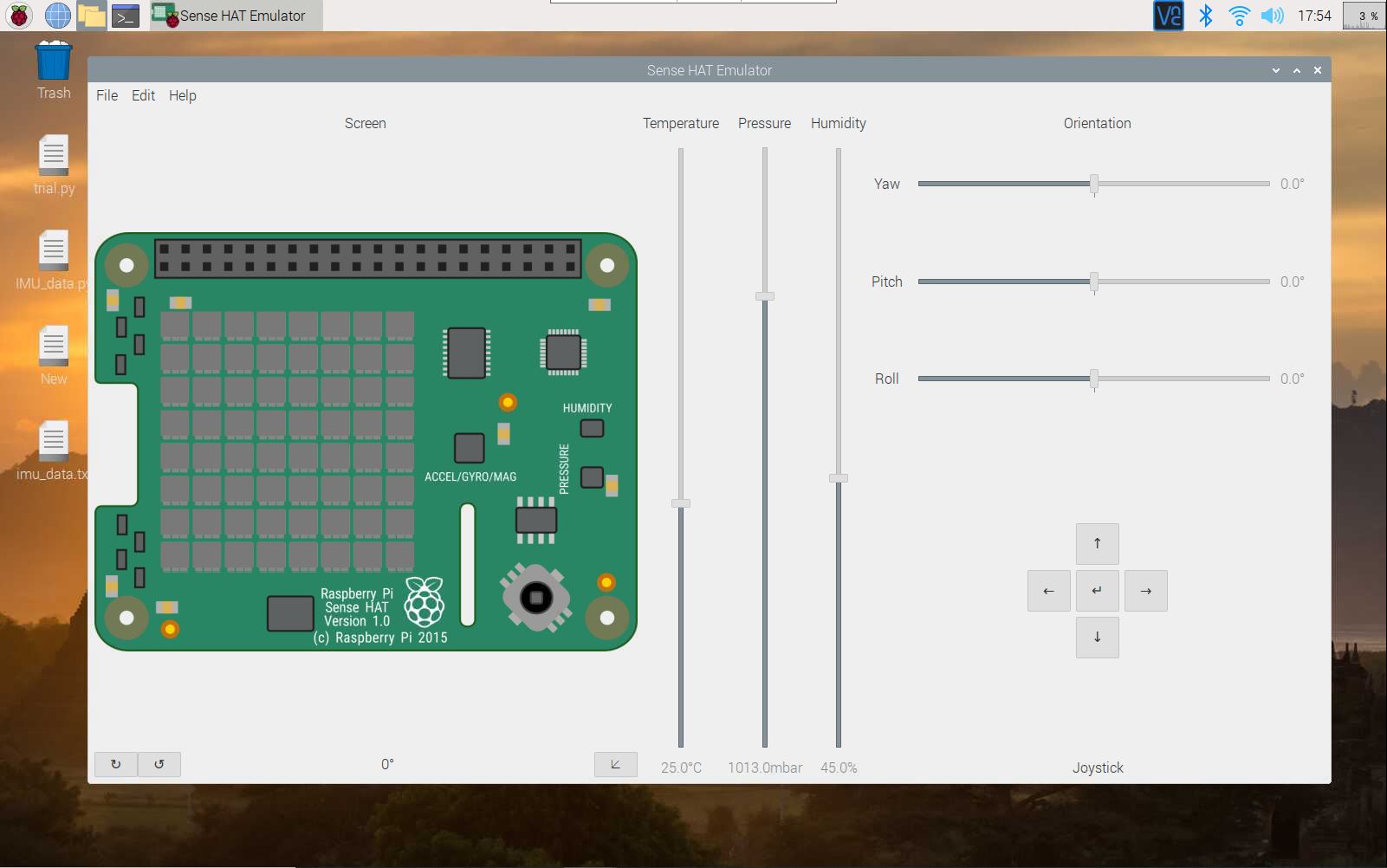
<https://docs.aws.amazon.com/iot/latest/developerguide/iot-gs-first-thing.html>

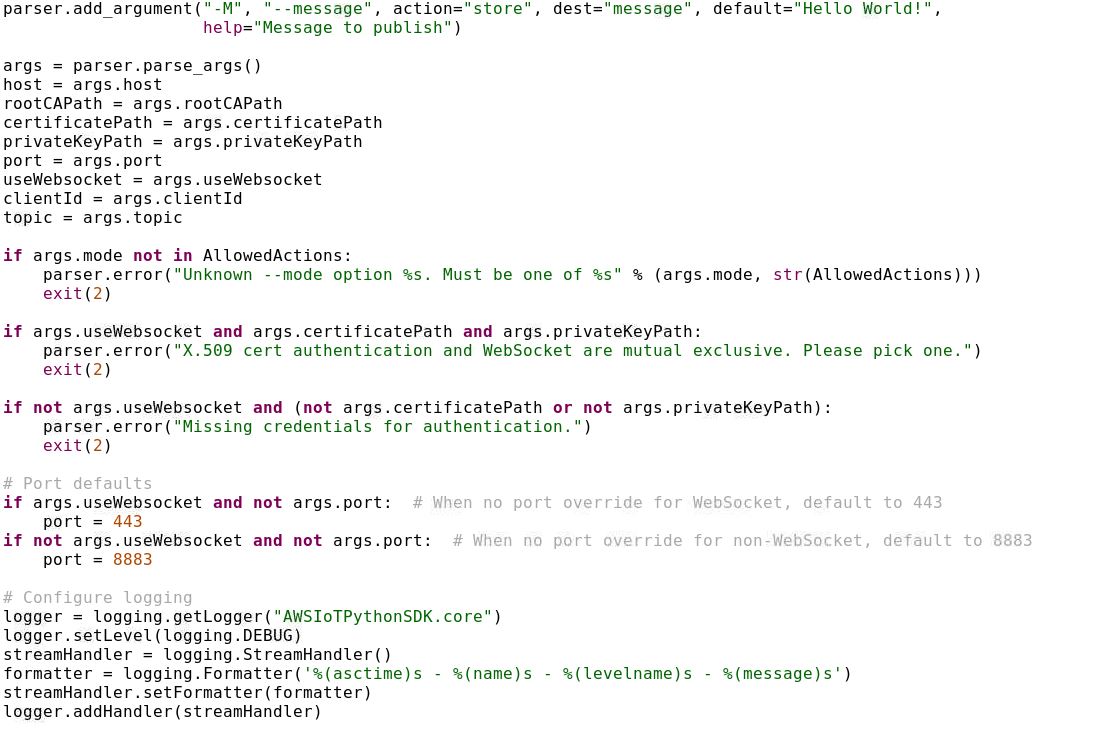
* After setting up the AWS IoT Core service, the next step to be done is to install the appropriate AWS SDK (Software Development Kit) on the Raspberry Pi in order to connect the IoT device to the cloud. The SDK used in this case is the AWS IoT Device SDK for Python. The procedure for the setup is explained in the AWS documentation below:

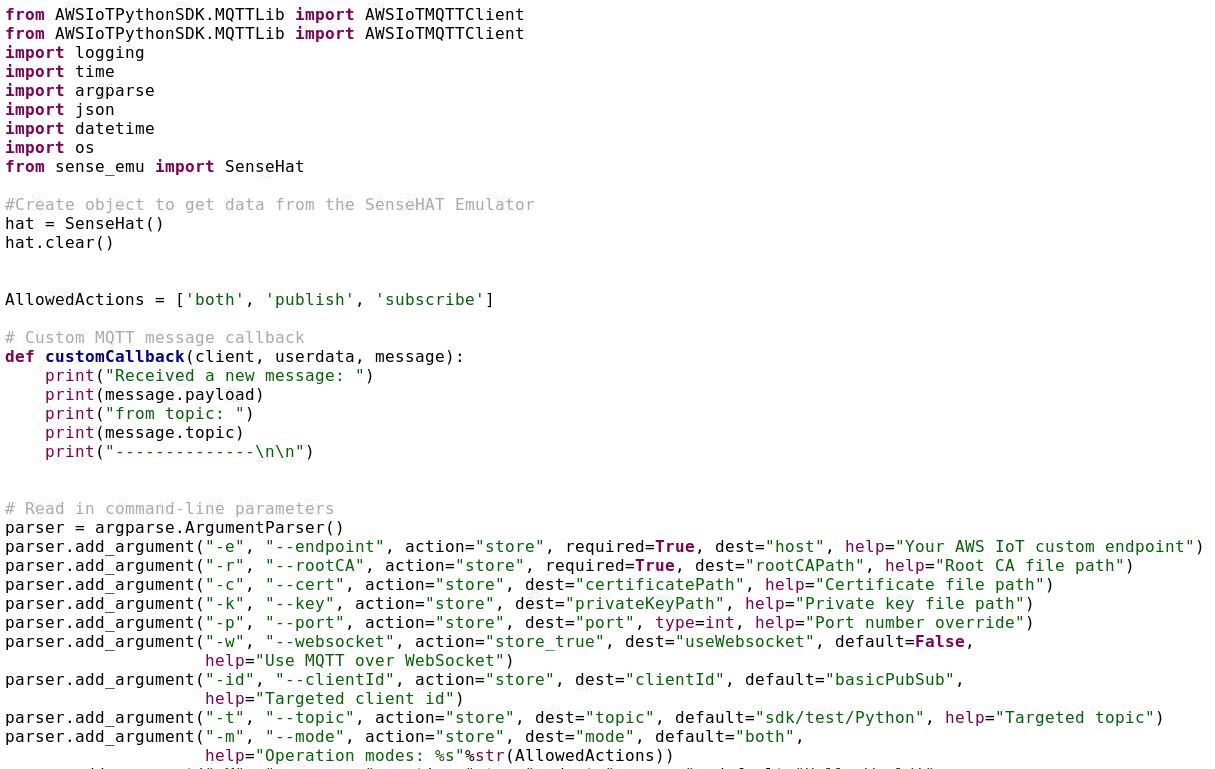
<https://docs.aws.amazon.com/greengrass/latest/developerguide/IoT-SDK.html>

1. **Data:**

* The data is sent using the MQTT protocol which is one of the protocols that are supported by AWS IoT Core.
* The data being sent is the magnetometer, accelerometer, and gyroscope data acquired from the SenseHat Emulator in the Raspberry Pi by manipulating the Yaw, Pitch and Roll scroll bars. The data is sent in 2 second intervals in the JSON format along with a timestamp.



1. **Code:**

****

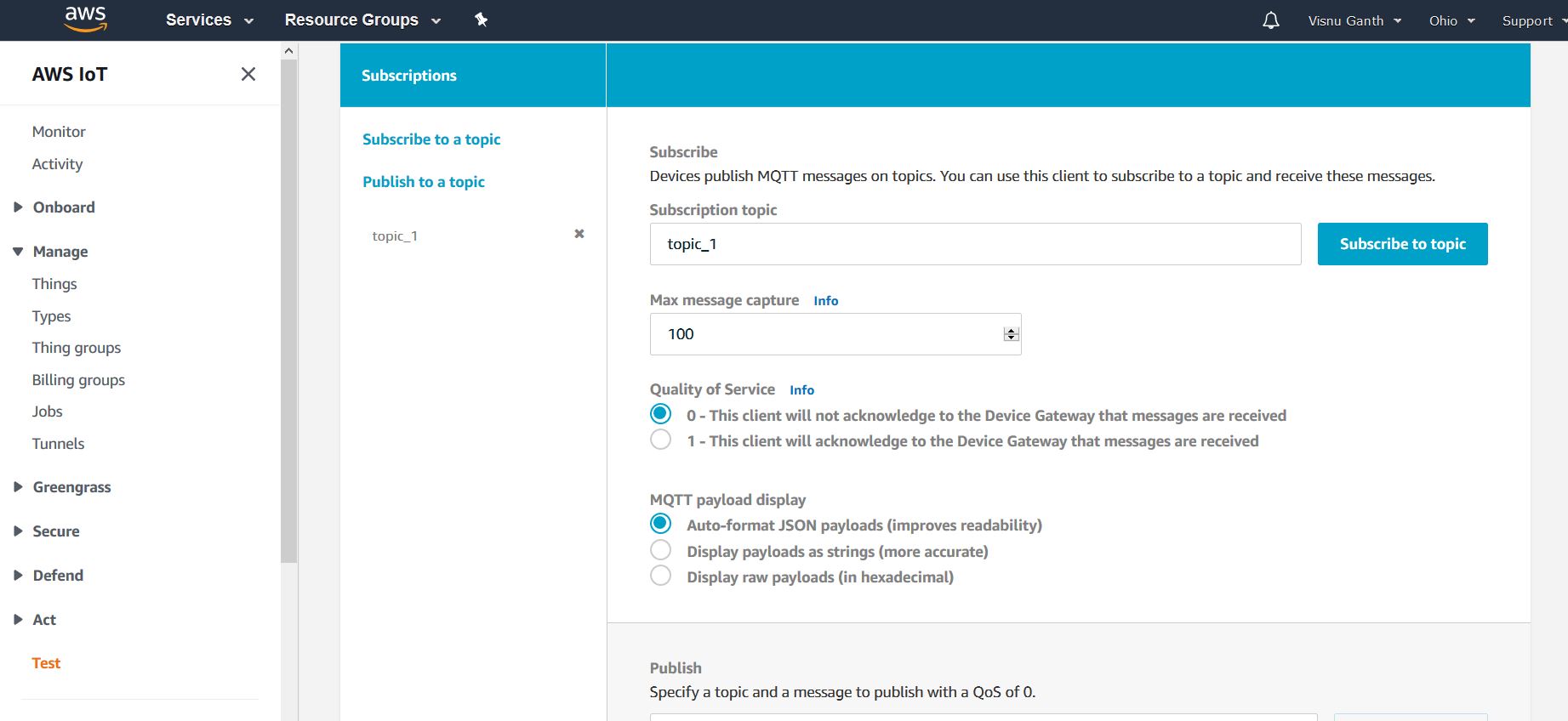
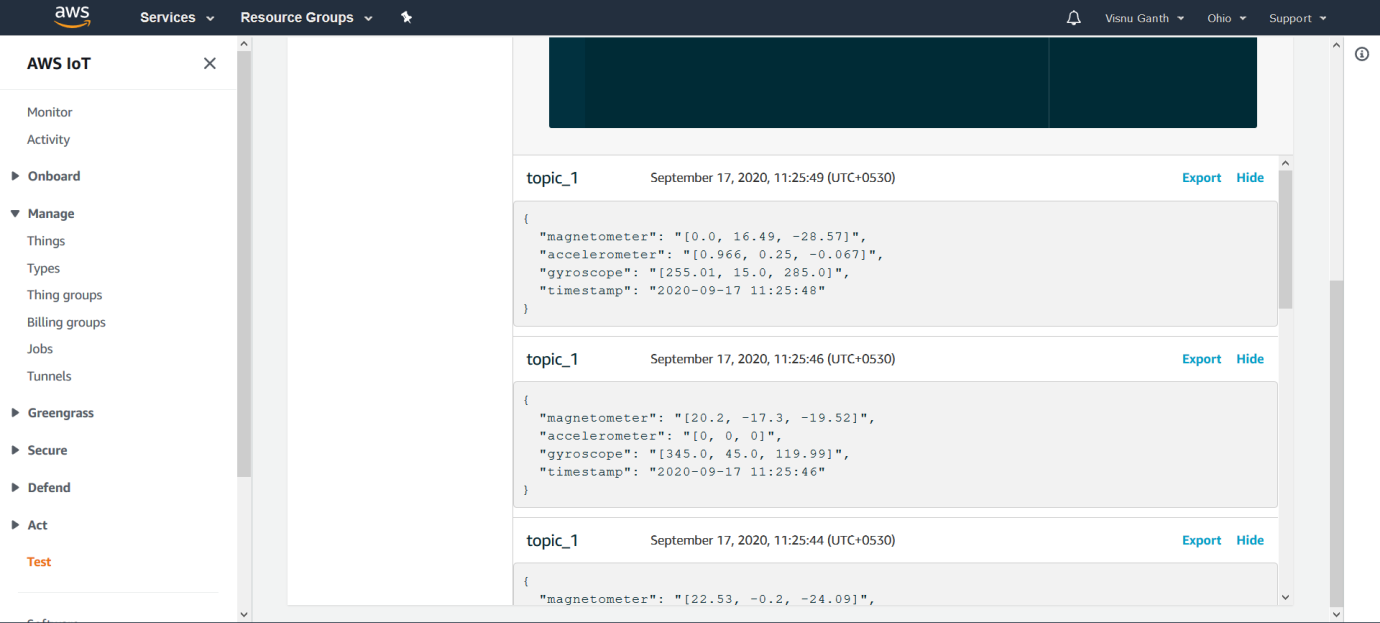
****

* The code is executed using the following command:

python3 filename.py -t topic\_1 -r root-certificate-path -c device-certificate-path -k private-key-path -m publish -e AWS-endpoint-address

* As soon as the code is executed, the SenseHat Emulatore pops up automatically.

1. **Data Received in AWS IoT Core:**

* The communication protocol used here is the MQTT protocol. This protocol follows a publisher subscriber model (the Raspberry Pi as the publisher and AWS IoT Core as the subscriber). The data is sent on a topic which acts as a channel for the data to pass through.
* ****The name of the topic here is topic\_1. So, in the AWS IoT Core must be subscribed to topic\_1.
* Once the topic\_1 channel has been subscribed to, we can see the data being sent from the Raspberry Pi in the JSON format on the AWS IoT Core Console.

**To set up the Rule:**

* Follow this link under “Create a rule with Lambda action” and ignore steps 10-15:

<https://docs.aws.amazon.com/iot/latest/developerguide/iot-lambda-rule.html#create-lambda-rule>