### **1. AWS IoT Core Setup**

I started by:

* I created the IoT thing named rpithing in AWS IoT Core to represent the Raspberry Pi.
* I downloaded the necessary security credentials, which include:
  + The **device certificate** (rpithing.cert.pem)
  + The **private key** (rpithing.private.key)
  + The **public key** (rpithing.public.key)
  + The **policy** (rpithing-Policy), which defines the actions my Raspberry Pi is allowed to perform in AWS IoT (like connecting, publishing, and subscribing to topics).

I downloaded the **Amazon Root CA certificate**, for TLS authentication between my Raspberry Pi and AWS IoT Core.

### **2. Installing Required Libraries**

On my Raspberry Pi, I installed the following:

* I used a Python library (AWS IoT SDK) to handle MQTT communication with AWS IoT.

### **3. Configuring MQTT Connection**

Next, I configured the MQTT connection by:

* **Setting the AWS IoT Endpoint**: I used the AWS IoT endpoint URL (a33ngcpde4nm4d-ats.iot.us-east-1.amazonaws.com) to connect the Raspberry Pi to the AWS IoT MQTT broker.
* **I configured** the MQTT client to use:
  + My **device certificate** and **private key** for authentication.
  + The **Amazon Root CA certificate** to authenticate AWS IoT Core’s endpoint.

### **4. Publishing and Subscribing to MQTT Topics**

To send and receive messages, I:

* **Published a Message**: I programmed the Raspberry Pi to publish messages to a topic (sdk/test/python) by using python programming language.
* **Subscribed to a Topic**: I configured the thing( rpithing ) on AWS IoT Core to listen for incoming message from the Raspberry Pi by subscribing to the same topic ( sdk/test/python ).

### **5. Message Display**

MQTT test client was used to subscribe to the topic and the message was displayed