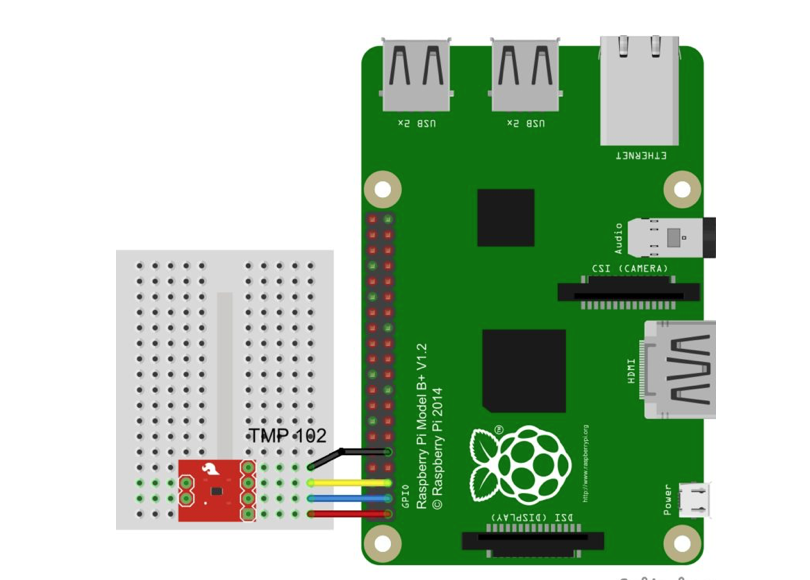
DIGIOTAI’S Step by step guide to Monitor the cold Storage temperature and humidity from DHT11 sensor with Raspberry Pi in a logistics environment in a supply chain management scenario

Step 1 Setting up RaspberryPi and Install Libraries

Raspberry Pi is a small computer that can use to learn programming supports python programming. The newer versions of RaspberryPi are supporting Windows 10 also.

RaspberryPi circuit



If Node.js is not installed, open the terminal and run the following commands

**$ sudo apt update**

**$ sudo apt full-upgrade**

**$ sudo apt-get install nodejs npm**

To verify installation of nodejs

**$ node -v**

**$ npm -v**

Setting up package.json file run the following commands

**$ npm init -y**

**$ npm install aws-iot-device-sdk --save**

**$ npm install rpi-dht-sensor --save**

Create a new file name called index.js at the root of the AWS-IOT-Thing folder and paste the below given contents in index.js

**var awsIot = require('aws-iot-device-sdk');**

**var rpiDhtSensor = require('rpi-dht-sensor');**

**var dht = new rpiDhtSensor.DHT11(2); // `2` => GPIO2**

**const NODE\_ID = 'Pi3-DHT11-Node';**

**const INIT\_DELAY = 15;**

**const TAG = '[' + NODE\_ID + '] >>>>>>>>> ';**

**console.log(TAG, 'Connecting...');**

**var thingShadow = awsIot.thingShadow({**

**keyPath: './certs/db80b0f635-private.pem.key',**

**certPath: './certs/db80b0f635-certificate.pem.crt',**

**caPath: './certs/RootCA-VeriSign-Class 3-Public-Primary-Certification-Authority-G5.pem',**

**clientId: NODE\_ID,**

**host: 'a1afizfoknpwqg.iot.us-east-1.amazonaws.com',**

**port: 8883,**

**region: 'us-east-1',**

**debug: false, // optional to see logs on console**

**});**

**thingShadow.on('connect', function() {**

**console.log(TAG, 'Connected.');**

**thingShadow.register(NODE\_ID, {}, function() {**

**console.log(TAG, 'Registered.');**

**console.log(TAG, 'Reading data in ' + INIT\_DELAY + ' seconds.');**

**setTimeout(sendData, INIT\_DELAY \* 1000); // wait for `INIT\_DELAY` seconds before reading the first record**

**});**

**});**

**function fetchData() {**

**var readout = dht.read();**

**var temp = readout.temperature.toFixed(2);**

**var humd = readout.humidity.toFixed(2);**

**return {**

**"temp": temp,**

**"humd": humd**

**};**

**}**

**function sendData() {**

**var DHT11State = {**

**"state": {**

**"desired": fetchData()**

**}**

**};**

**console.log(TAG, 'Sending Data..', DHT11State);**

**var clientTokenUpdate = thingShadow.update(NODE\_ID, DHT11State);**

**if (clientTokenUpdate === null) {**

**console.log(TAG, 'Shadow update failed, operation still in progress');**

**} else {**

**console.log(TAG, 'Shadow update success.');**

**}**

**// keep sending the data every 30 seconds**

**console.log(TAG, 'Reading data again in 30 seconds.');**

**setTimeout(sendData, 30000); // 30,000 ms => 30 seconds**

**}**

**thingShadow.on('status', function(thingName, stat, clientToken, stateObject) {**

**console.log('received ' + stat + ' on ' + thingName + ':', stateObject);**

**});**

**thingShadow.on('delta', function(thingName, stateObject) {**

**console.log('received delta on ' + thingName + ':', stateObject);**

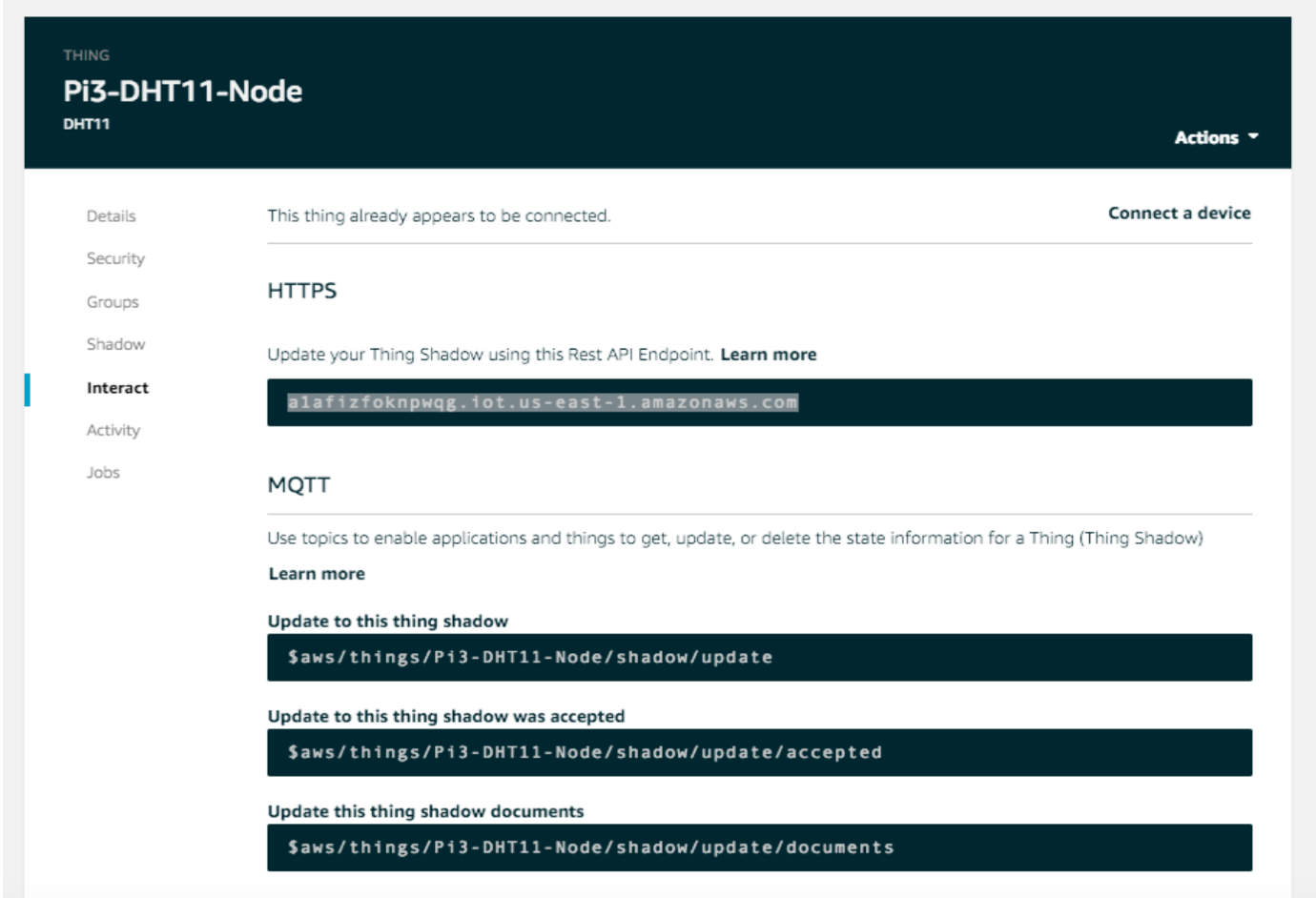
**});**

**thingShadow.on('timeout', function(thingName, clientToken) {**

**console.log('received timeout on ' + thingName + ' with token:', clientToken);**

**});**

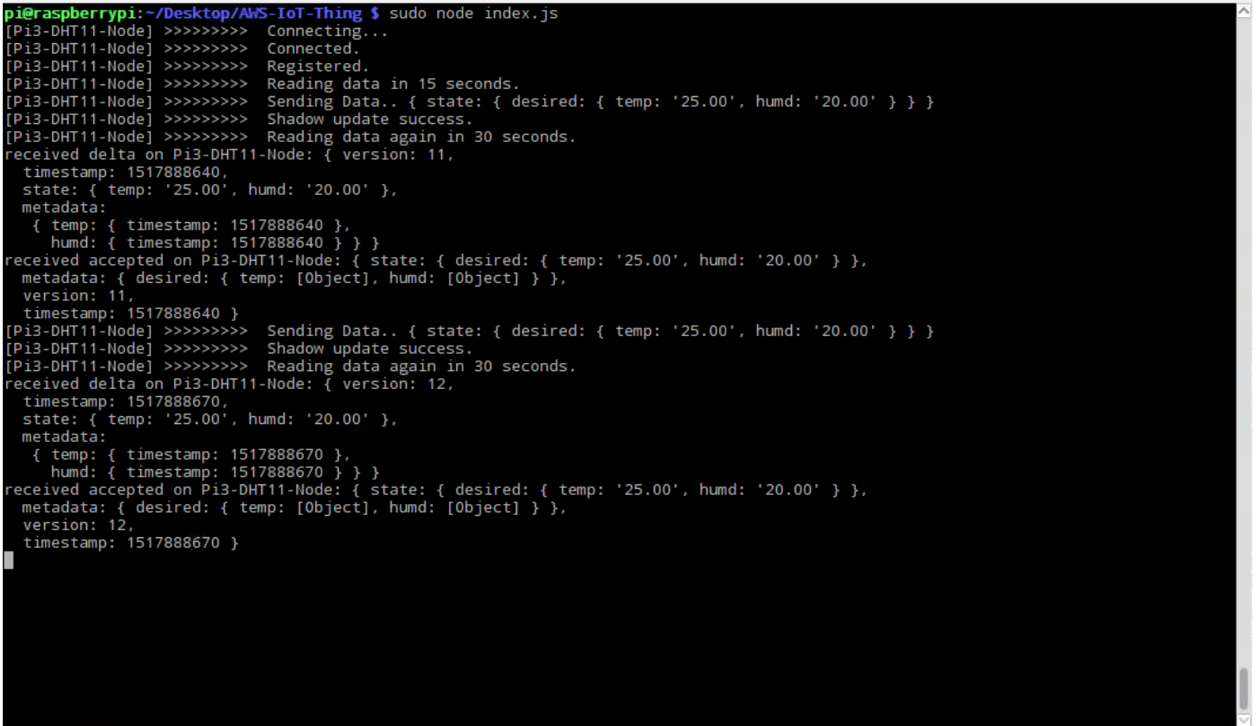
Note: Don’t forget to change the host name as per your amazon aws account



Create the new folder called certs in aws-iot-thing folder and paste the downloaded certificates later from amazon aws Account.

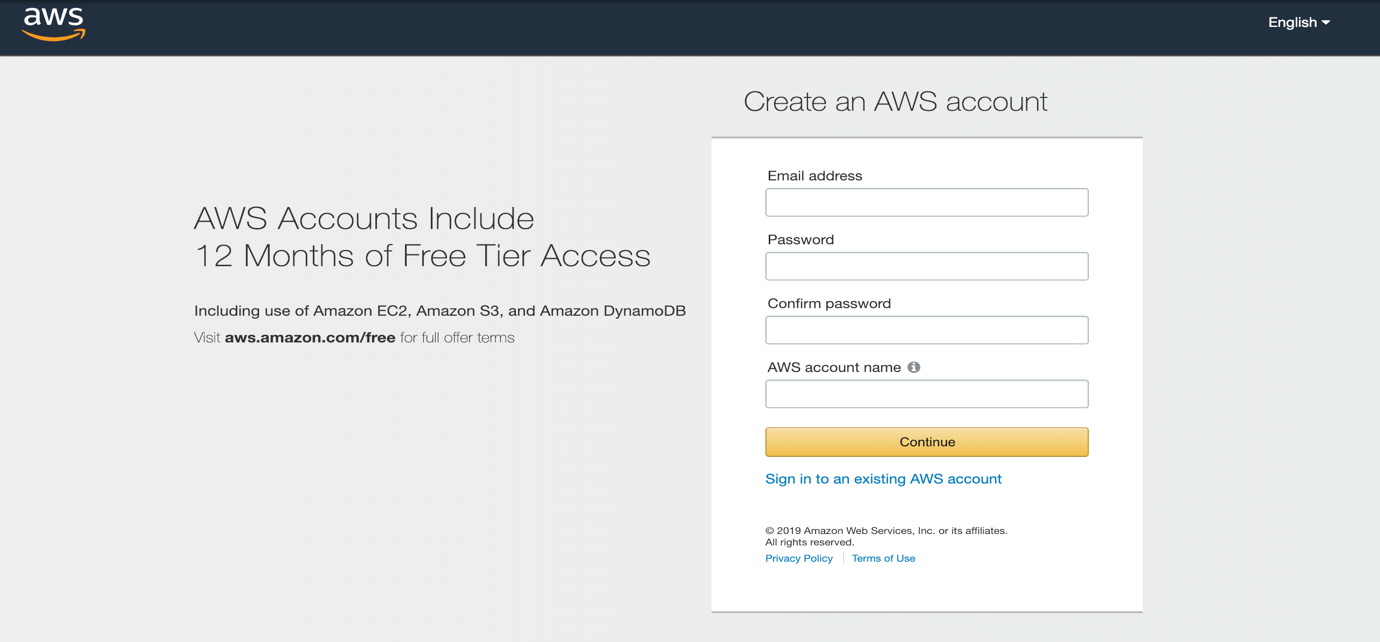
Save the file and execute following command

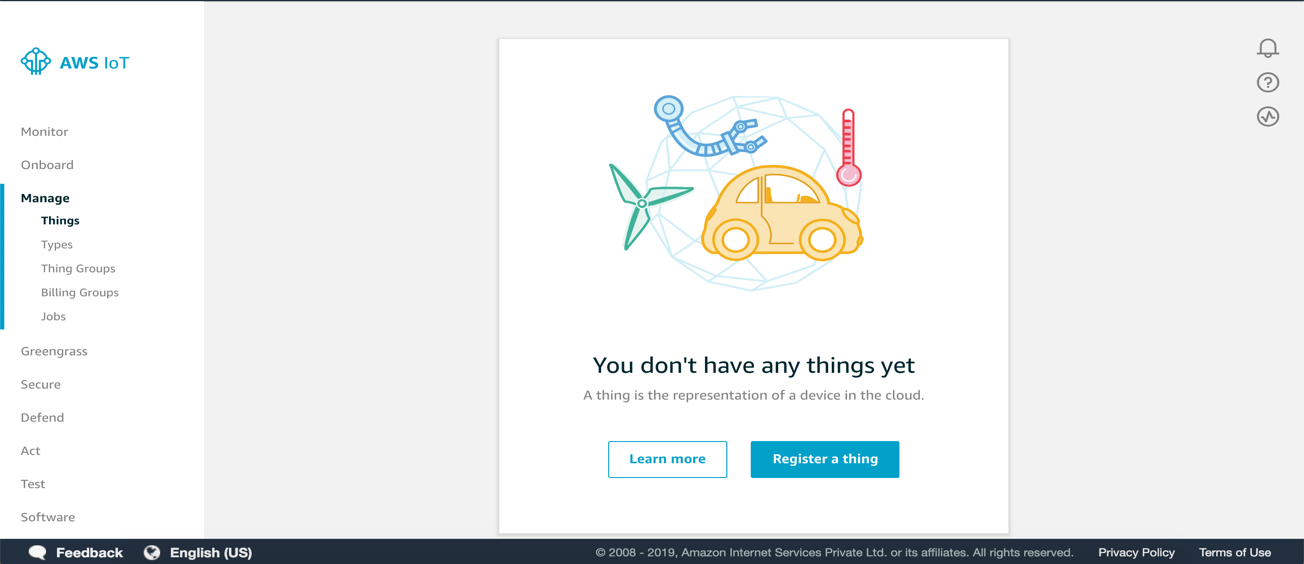
**$ sudo node index.js**

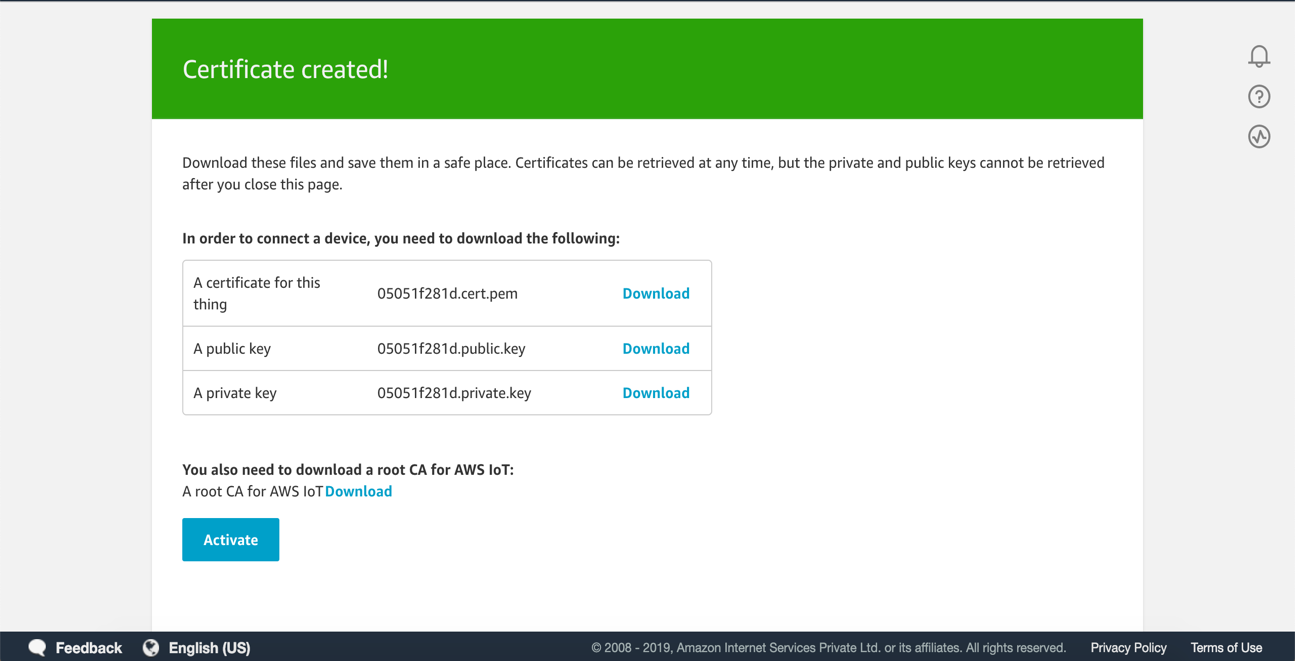


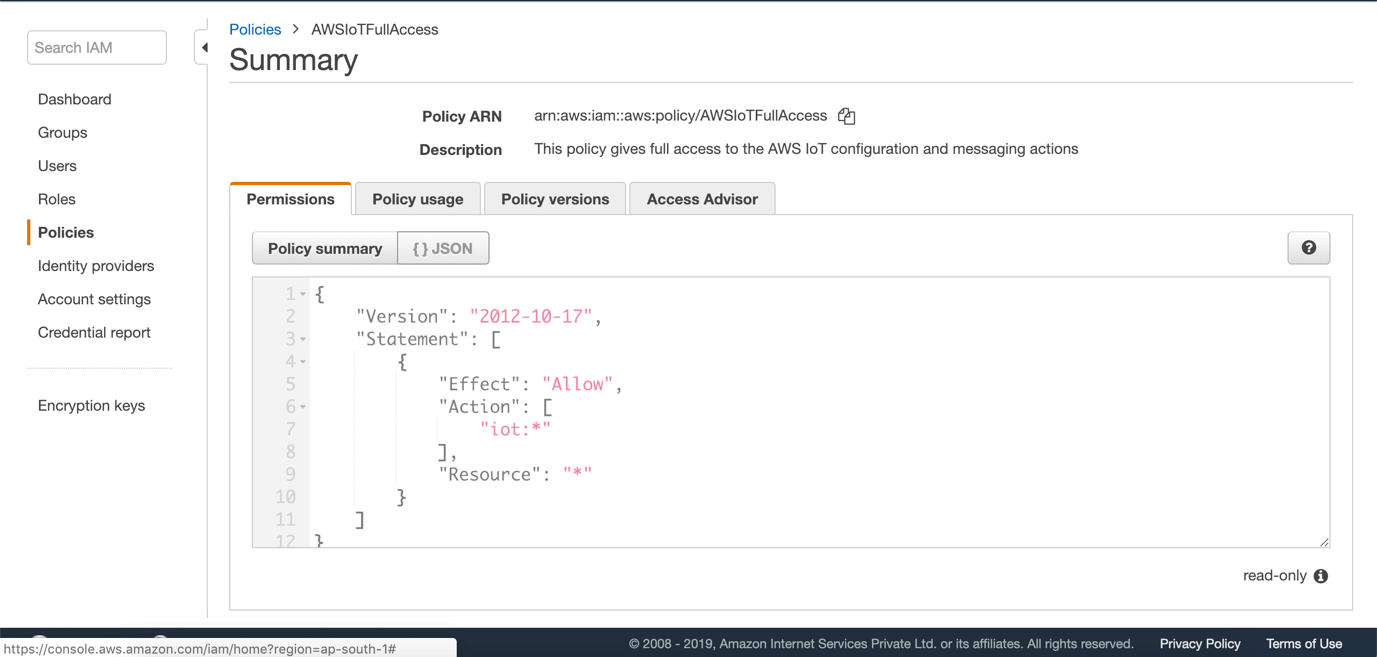
Step 2: Setting up Amazon AWS

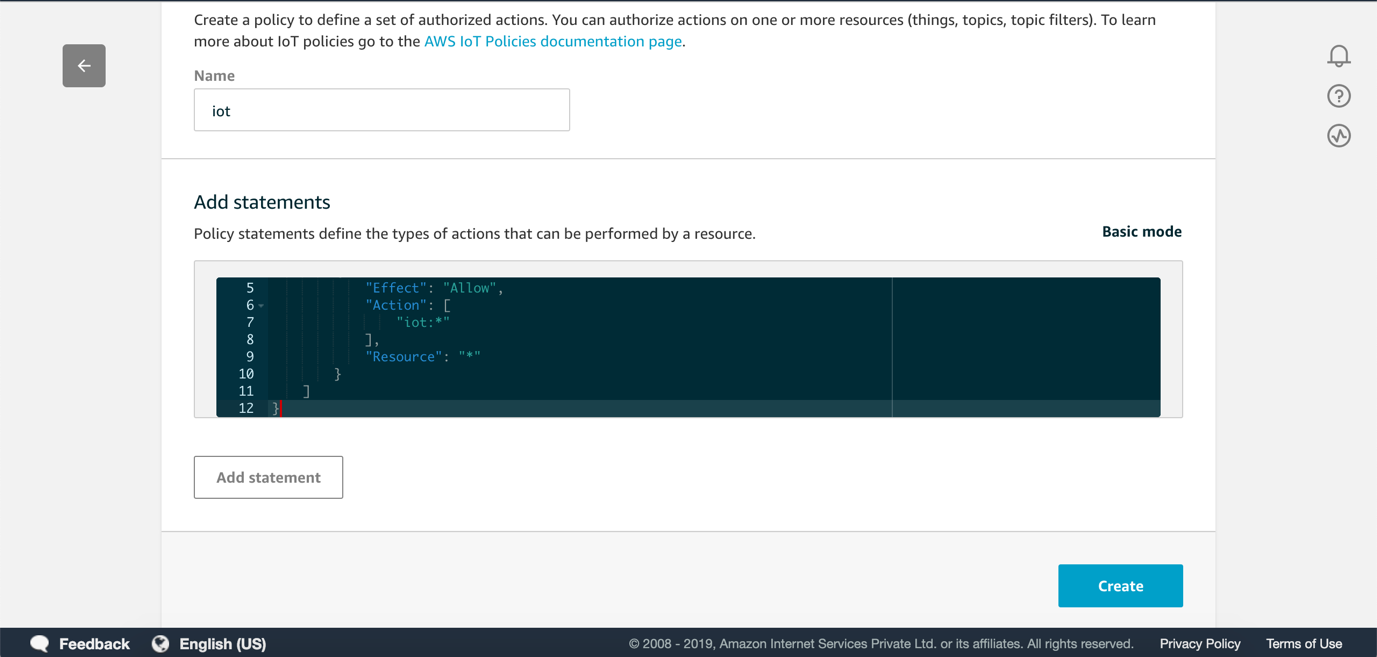
1. Go to [console.aws.amazon.com](http://console.aws.amazon.com/) and Sign up a free account



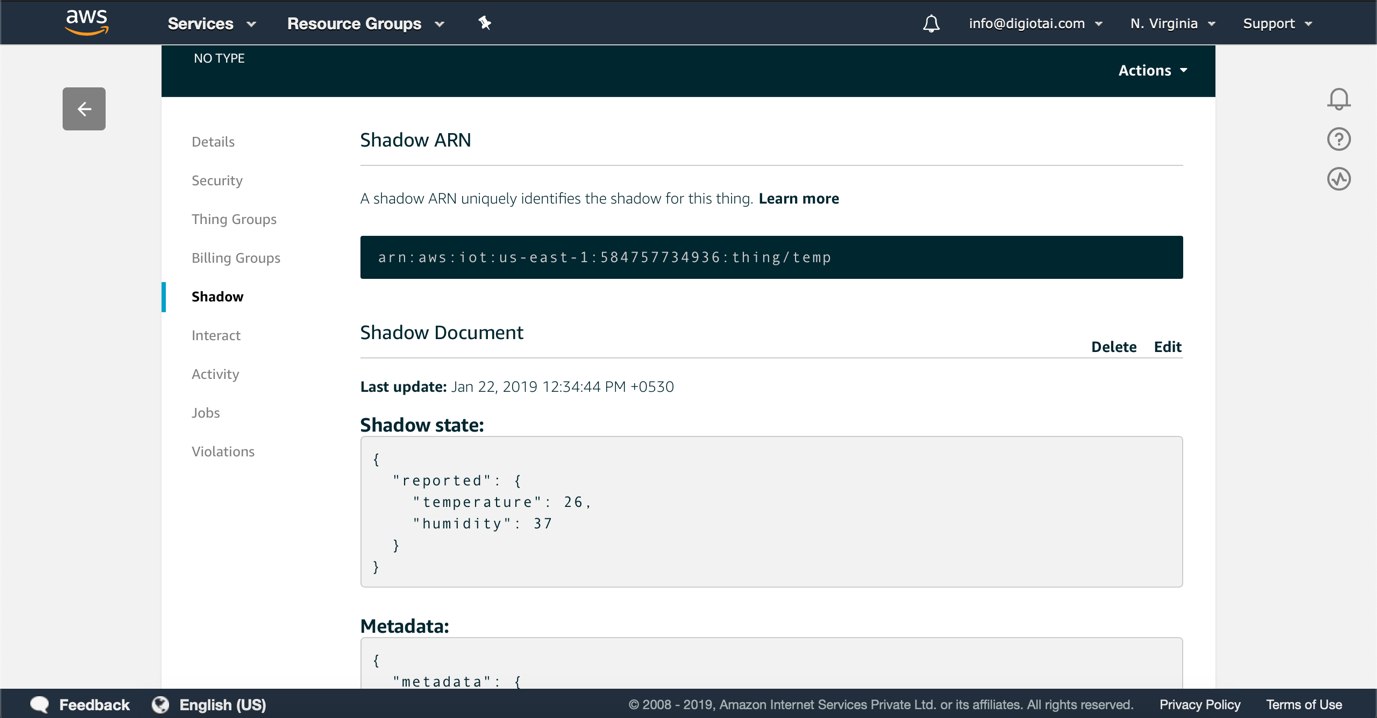
1. Register a thing in IoT Core Services
2. Create certificate and Activate it

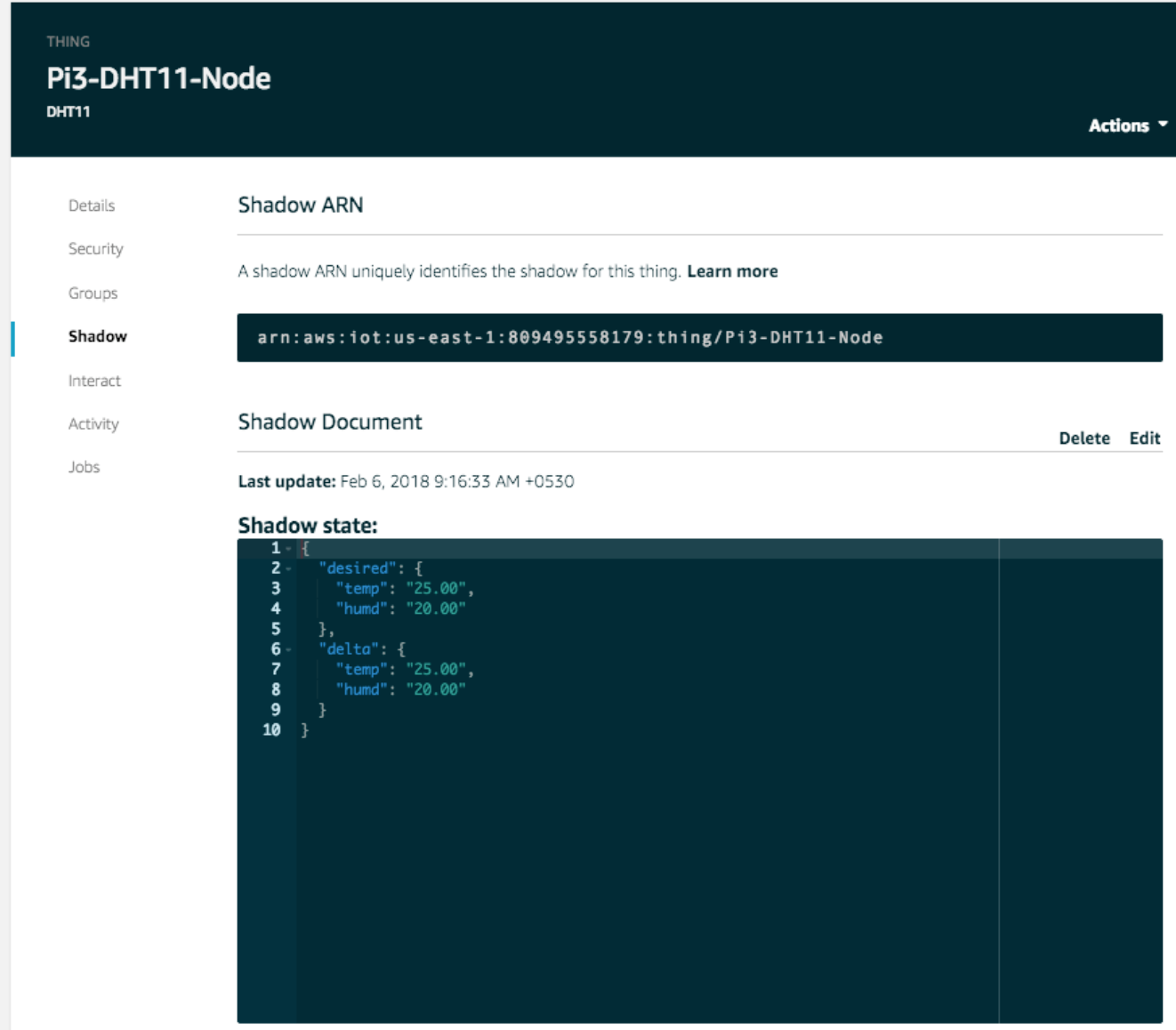


1. Go to IAM policies and search for AWSIoTFullAccess and copy the json code and paste it in IoT Core Policies Create a new policy and click on advanced and paste the code in it and click on create .



1. Go back to thing creation and attach a policy created to it and register a thing.





AWS credentials: info@digiotai.com