

CIS 600
Internet Of Things : Application Development
Spring 2025

Name : Venkata Sri Siva Ramakrishna Palaparthys

SUID : 433976193

NetID : vpalapar

Date : 04 - April - 2025

In Class Exercise - Week 10

Brief Explanation of the Steps Used in Developing the IoT System :

1. Creation of an IoT Thing in AWS IoT Core :

- Started with creating a new IoT thing in the AWS IoT Core console, which automatically generated a unique certificate and associated security credentials including a private key, public key, and root CA files.

2. Downloading the Required Credentials :

- Security credentials such as AmazonRootCA1, the certificate, and the private key were downloaded and stored for use in the simulation script.

3. Python MQTT Script Configuration :

- Wrote a python script to simulate environmental sensor data such as temperature, humidity, and CO2 levels. The script includes the endpoint taken from AWS and local paths to the downloaded certificates and key files.

4. Testing MQTT Communication :

- The AWS IoT MQTT Test Client was used to verify that the Python script was correctly publishing data to the topic update/environment/dht1. Real-time data flow was confirmed by observing the incoming messages in AWS IoT MQTT Test Client.

5. Setting Up an S3 Bucket :

- An Amazon S3 bucket (my-s3-bucket-iot) was created to serve as the final storage location for the incoming sensor data.

6. Creating a Kinesis Firehose Stream :

- A Firehose delivery stream was configured to accept data via the Direct PUT method and deliver it to the S3 bucket. Basic settings were selected, with no data transformation or compression enabled.

7. Creating an IoT Rule :

- An AWS IoT rule was configured to forward messages from the MQTT topic update/environment/dht1 to the Firehose delivery stream. An appropriate IAM role was created to allow this connection.

8. Confirming Data Flow and Storage :

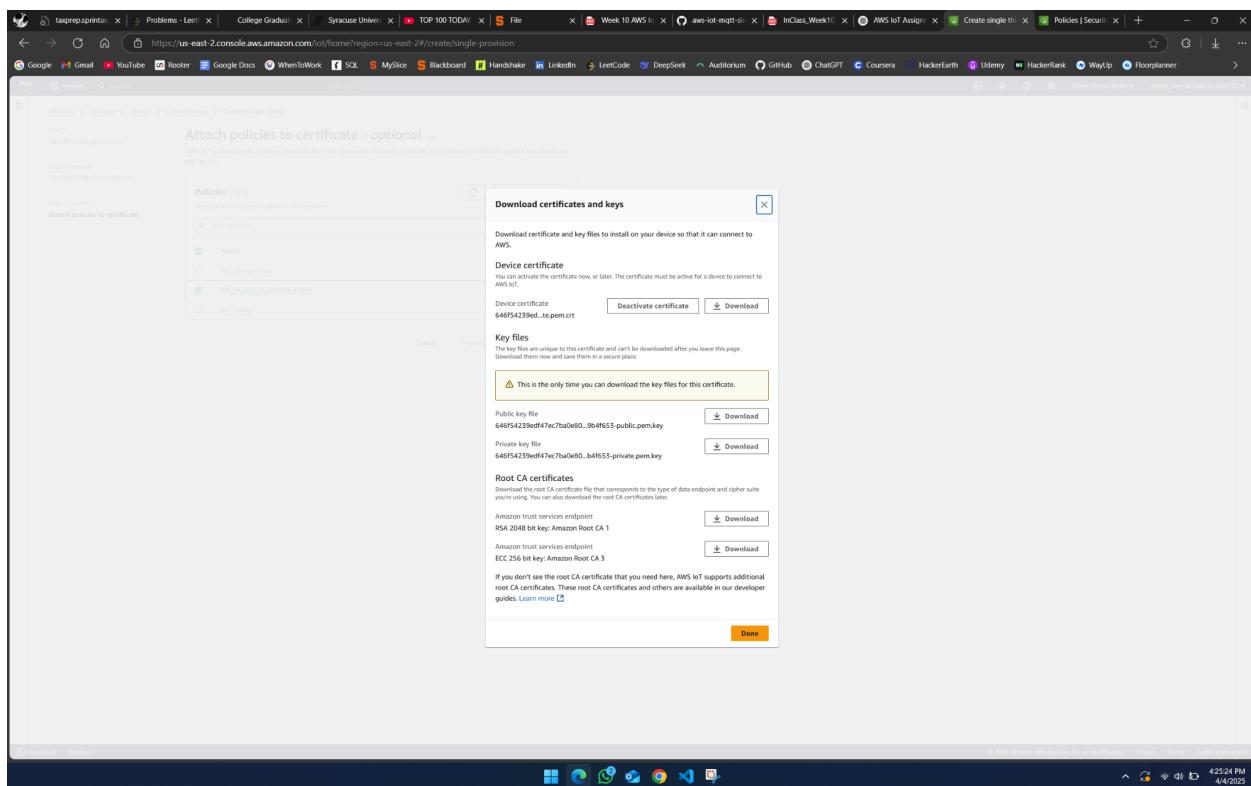
- The Python script was run again to push data. Monitoring tools in AWS Firehose confirmed successful delivery, and the presence of JSON files in the S3 bucket validated that data storage was functioning as expected.

9. Verifying Output :

- Files downloaded from the S3 bucket contained the expected JSON-formatted sensor data, confirming that the entire IoT data pipeline was functioning correctly.

Screenshots :

Creating a Thing in AWS IoT Core and Setting Up AWS IoT Certificates and Policies :



The screenshot shows the AWS IoT Things management interface. A success message at the top states "You successfully created certificate 646f54239edf47ec7bae00ced1a65d7400a1032a116f5d0042687a04f553." Below this, the "Things" section lists three items: "DHT11_Simulator", "IoT_Thing", and "Name". The sidebar contains links for Monitor, Connect, Test, Manage, and other AWS services.

The screenshot shows the "DHT11_Simulator" thing details page. It includes sections for "Thing details" (Name: DHT11_Simulator, ARN: arn:aws:iot:us-east-2:503561455720:thing/DHT11_Simulator), "Attributes" (empty), and "Propagating attributes" (empty). The sidebar is identical to the previous screenshot.

The screenshot shows the AWS IoT console's MQTT test client interface. On the left, a sidebar lists various AWS services like Monitor, Connect, Test, Manage, Device software, and Feature spotlight. The main area displays four MQTT messages sent to the topic `update/environment/dht1`. Each message contains the following JSON payload:

```
{
  "thingId": "dht1",
  "temperature": 35,
  "humidity": 48,
  "co2": 770,
  "datetime": "2025-04-04T20:53:21.996Z"
}
```

The messages were received at different times: April 04, 2025, 16:53:22 (UTC-0400), April 04, 2025, 16:53:20 (UTC-0400), April 04, 2025, 16:53:18 (UTC-0400), and April 04, 2025, 16:53:16 (UTC-0400).

Creating and Configuring the Amazon S3 Bucket :

The screenshot shows the AWS S3 console's buckets page. On the left, a sidebar lists General purpose buckets, Storage Lens, and AWS Marketplace for S3. The main area shows a success message: "Successfully created bucket 'my-s3-bucket-iot'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, an account snapshot is displayed, and a table lists the general purpose bucket "my-s3-bucket-iot".

Name	AWS Region	IAM Access Analyzer	Creation date
my-s3-bucket-iot	US East (Ohio) us-east-2	View analyzer for us-east-2	April 4, 2025, 17:03:06 (UTC-04:00)

Creating Kinesis Data Firehose Stream to S3 :

iot-firehose-stream was successfully created.

iot-firehose-stream

Firehose stream details

- Status: Active
- Destination: Amazon S3
- ARN: arn:aws:firehose:us-east-2:2505561455720:deliverystream/iot-firehose-stream
- Data transformation: Not enabled
- Source: Direct PUT
- Dynamic partitioning: Not enabled
- Creation time: April 04, 2025 at 17:08 EDT
- Error logs status: 0 Destination error logs

Test with demo data (Info) Ingest simulated data to test the configuration of your Firehose stream. Standard Amazon Data Firehose charges apply.

Monitoring | Configuration | Destination error logs

Firehose stream metrics

1h 3h 12h 1d 3d 1w Custom UTC timezone Add to dashboard

Metric	Value	Description
Incoming bytes	0.0	No data available. Try adjusting the dashboard time range.
Incoming put requests	0.0	No data available. Try adjusting the dashboard time range.
Incoming records	0.0	No data available. Try adjusting the dashboard time range.
Throttled records (Count)	0.0	No data available. Try adjusting the dashboard time range.
Delivery to Amazon S3 success	0.0	No data available. Try adjusting the dashboard time range.
Delivery to Amazon S3 data freshness (Maximum)	0.0	No data available. Try adjusting the dashboard time range.
Records delivered to Amazon S3 (Sum)	0.0	No data available. Try adjusting the dashboard time range.
Bytes delivered to Amazon S3 (Sum)	0.0	No data available. Try adjusting the dashboard time range.

iot-firehose-stream

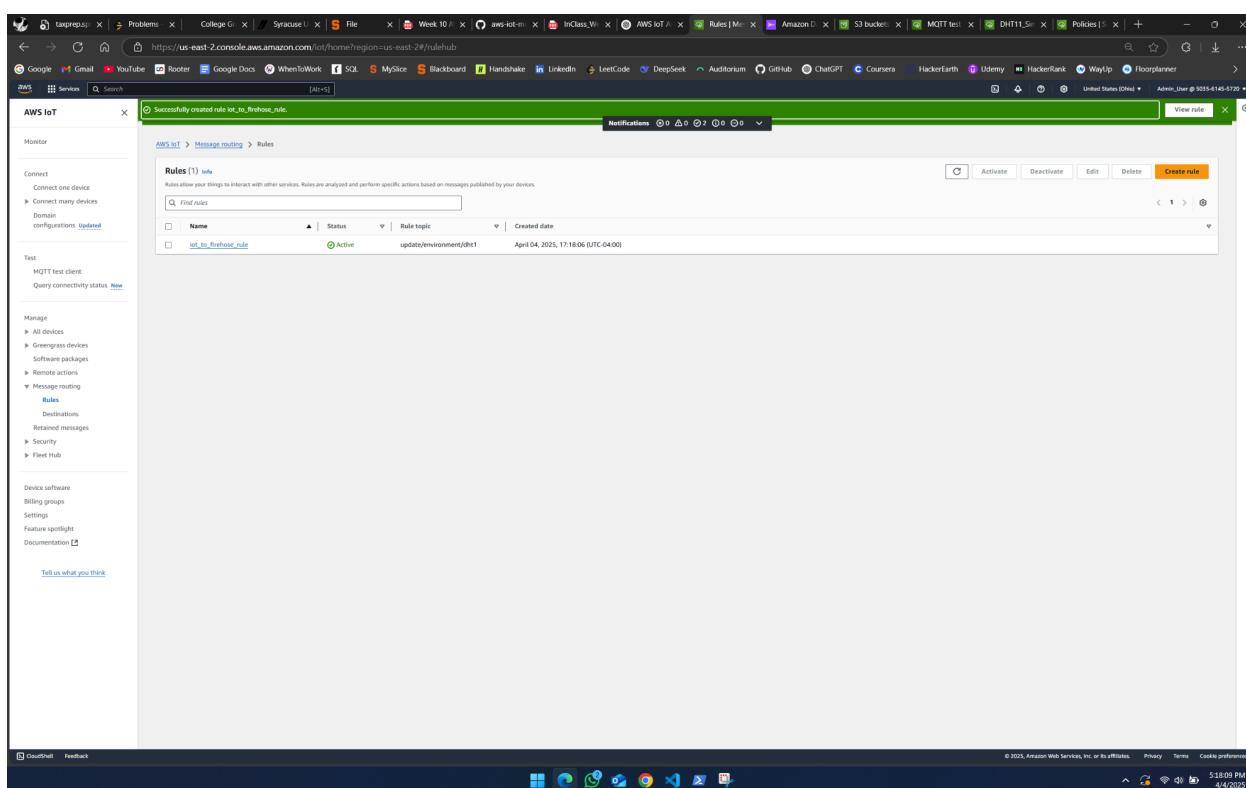
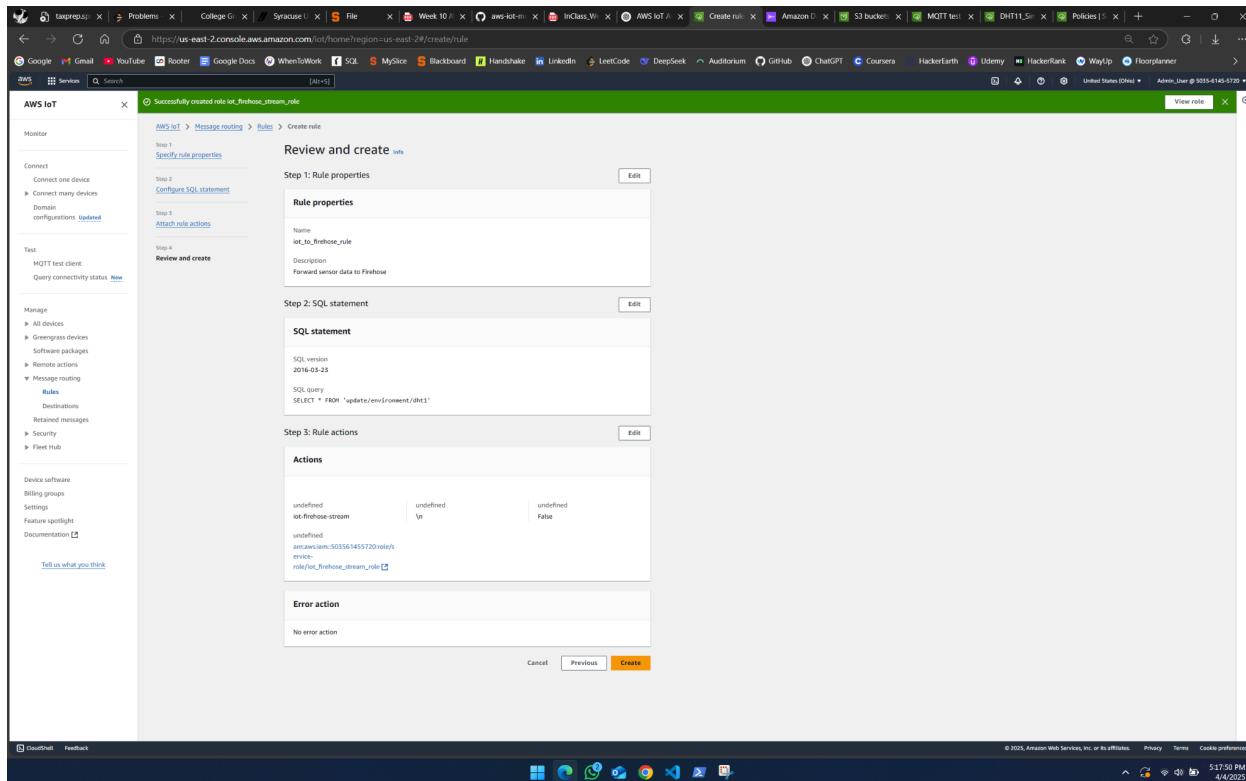
Monitoring | Configuration | Destination error logs

Firehose stream metrics

1h 3h 12h 1d 3d 1w Custom UTC timezone Add to dashboard

Metric	Value	Description
Incoming bytes	0.0	No data available. Try adjusting the dashboard time range.
Incoming put requests	0.0	No data available. Try adjusting the dashboard time range.
Incoming records	0.0	No data available. Try adjusting the dashboard time range.
Throttled records (Count)	0.0	No data available. Try adjusting the dashboard time range.
Delivery to Amazon S3 success	0.0	No data available. Try adjusting the dashboard time range.
Delivery to Amazon S3 data freshness (Maximum)	0.0	No data available. Try adjusting the dashboard time range.
Records delivered to Amazon S3 (Sum)	0.0	No data available. Try adjusting the dashboard time range.
Bytes delivered to Amazon S3 (Sum)	0.0	No data available. Try adjusting the dashboard time range.

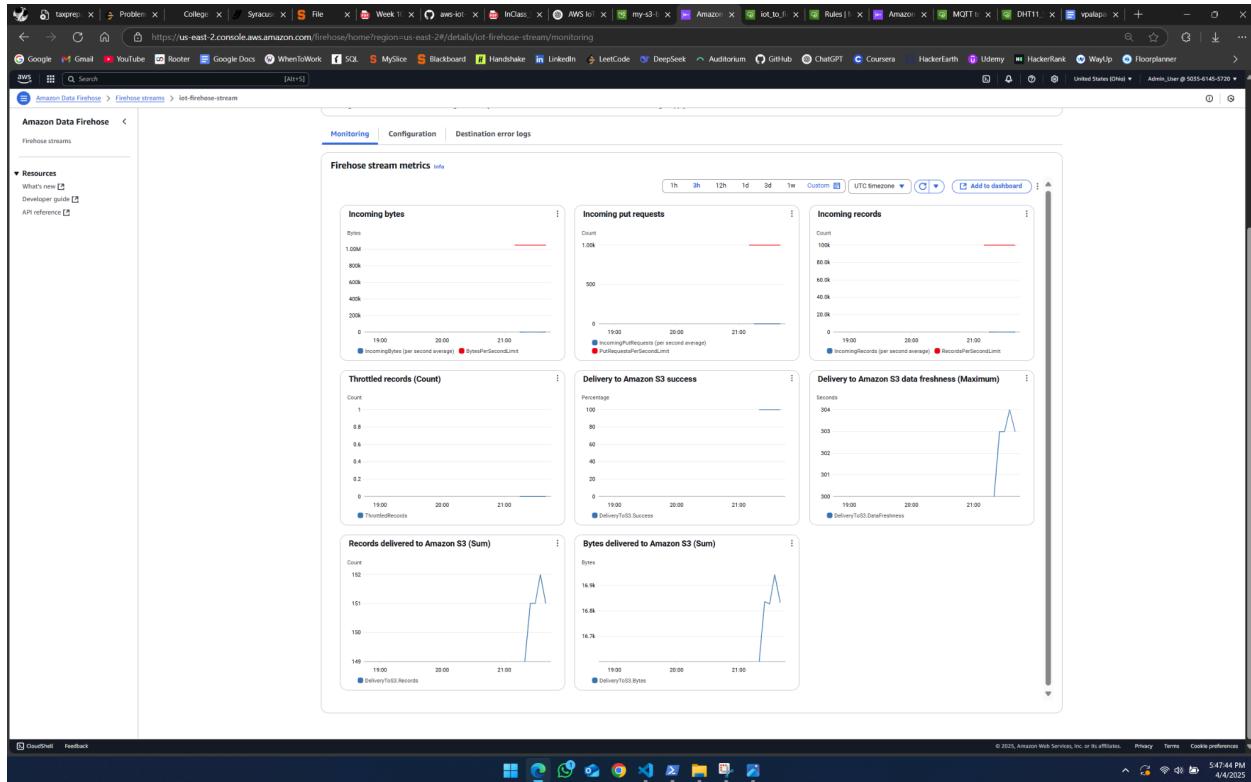
Creating IoT Rule to Forward MQTT Data to Firehose - Completing IoT Rule Setup and Activating Data Pipeline :



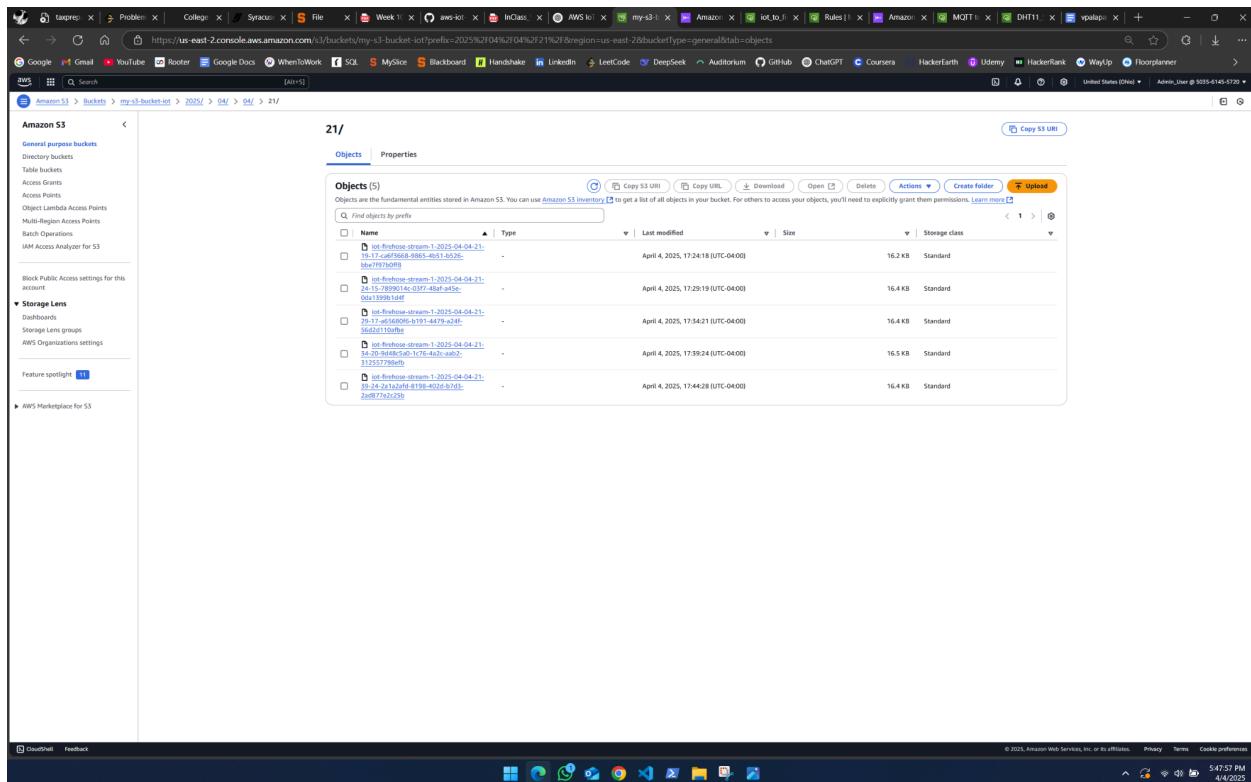
Re-running Python Script to Send Data :

```
PS C:\Users\Kris\Desktop\InClass\Week10> python mqtt_aws_simulator.py
AWS Client ID: pkhu-client_id-clientId
Topic: pubh.ClientId-clientId
Publishing: ("thingid": "dht1", "temperature": 37, "humidity": 43, "co2": 577, "datetime": "2025-04-04T21:19:16.897488")
Connected to AWS IoT with result code: 0
Publishing: ("thingid": "dht1", "temperature": 32, "humidity": 38, "co2": 400, "datetime": "2025-04-04T21:19:18.899298")
Message received-> update/environment/dht1: {"thingid": "dht1", "temperature": 34, "humidity": 50, "co2": 550, "datetime": "2025-04-04T21:19:18.899298"} 
Publishing: ("thingid": "dht1", "temperature": 26, "humidity": 74, "co2": 435, "datetime": "2025-04-04T21:19:20.901408")
Message received-> update/environment/dht1: {"thingid": "dht1", "temperature": 26, "humidity": 74, "co2": 435, "datetime": "2025-04-04T21:19:20.901408"} 
Publishing: ("thingid": "dht1", "temperature": 20, "humidity": 70, "co2": 907, "datetime": "2025-04-04T21:19:22.902518")
Message received-> update/environment/dht1 b: {"thingid": "dht1", "temperature": 20, "humidity": 70, "co2": 907, "datetime": "2025-04-04T21:19:22.902518"} 
Publishing: ("thingid": "dht1", "temperature": 27, "humidity": 80, "co2": 433, "datetime": "2025-04-04T21:19:24.903518")
Message received-> update/environment/dht1: {"thingid": "dht1", "temperature": 27, "humidity": 80, "co2": 433, "datetime": "2025-04-04T21:19:24.903518"} 
Publishing: ("thingid": "dht1", "temperature": 36, "humidity": 47, "co2": 687, "datetime": "2025-04-04T21:19:26.904501")
Message received-> update/environment/dht1: {"thingid": "dht1", "temperature": 36, "humidity": 47, "co2": 687, "datetime": "2025-04-04T21:19:26.904501"} 
Publishing: ("thingid": "dht1", "temperature": 21, "humidity": 60, "co2": 525, "datetime": "2025-04-04T21:19:28.907852")
Message received-> update/environment/dht1 b: {"thingid": "dht1", "temperature": 21, "humidity": 60, "co2": 525, "datetime": "2025-04-04T21:19:28.907852"} 
Publishing: ("thingid": "dht1", "temperature": 37, "humidity": 58, "co2": 558, "datetime": "2025-04-04T21:19:30.910472")
Message received-> update/environment/dht1 b: {"thingid": "dht1", "temperature": 21, "humidity": 58, "co2": 558, "datetime": "2025-04-04T21:19:30.910472"} 
Publishing: ("thingid": "dht1", "temperature": 37, "humidity": 58, "co2": 558, "datetime": "2025-04-04T21:19:32.912488")
Message received-> update/environment/dht1: {"thingid": "dht1", "temperature": 37, "humidity": 58, "co2": 558, "datetime": "2025-04-04T21:19:32.912488"} 
Publishing: ("thingid": "dht1", "temperature": 26, "humidity": 74, "co2": 435, "datetime": "2025-04-04T21:19:32.912488")
Message received-> update/environment/dht1 b: {"thingid": "dht1", "temperature": 26, "humidity": 74, "co2": 435, "datetime": "2025-04-04T21:19:32.912488"} 
Publishing: ("thingid": "dht1", "temperature": 20, "humidity": 70, "co2": 907, "datetime": "2025-04-04T21:19:32.912488")
```

Confirming Firehose Delivery Metrics and Success :



S3 Bucket Showing Delivered IoT Data Files :



Inspecting Metadata of an S3 Data Object :

The screenshot shows the AWS S3 Object Details page for the object 'iot-firehose-stream-1-2025-04-04-21-19-17-ca6f3668-9865-4b51-b526-bbe7f97b0ff8'. The object was created by the user 'ec14ed94d4dd7ee2af158eb4ff56c6216da775544acb7ff523def9f06e101' in the 'us-east-2' region on April 4, 2025, at 17:24:18 UTC. The file size is 16.2 KB and the type is a JSON file ('key'). The object has no bucket versioning enabled.

Viewing JSON Data from Downloaded S3 Object :

```
File Edit View
iot-firehose-stream-1-2025-04-04 + ...
{"thingid": "dht1", "temperature": 37, "humidity": 43, "co2": 577, "datetime": "2025-04-04T21:19:16.897488"}, {"thingid": "dht1", "temperature": 22, "humidity": 34, "co2": 506, "datetime": "2025-04-04T21:19:18.899298"}, {"thingid": "dht1", "temperature": 26, "humidity": 74, "co2": 435, "datetime": "2025-04-04T21:19:20.899148"}, {"thingid": "dht1", "temperature": 29, "humidity": 90, "co2": 493, "datetime": "2025-04-04T21:19:23.899158"}, {"thingid": "dht1", "temperature": 47, "humidity": 80, "co2": 593, "datetime": "2025-04-04T21:19:24.899169"}, {"thingid": "dht1", "temperature": 36, "humidity": 47, "co2": 687, "datetime": "2025-04-04T21:19:26.994894"}, {"thingid": "dht1", "temperature": 39, "humidity": 60, "co2": 625, "datetime": "2025-04-04T21:19:28.997852"}, {"thingid": "dht1", "temperature": 21, "humidity": 37, "co2": 558, "datetime": "2025-04-04T21:19:30.010472"}, {"thingid": "dht1", "temperature": 29, "humidity": 47, "co2": 1351, "datetime": "2025-04-04T21:19:32.912488"}, {"thingid": "dht1", "temperature": 39, "humidity": 44, "co2": 1059, "datetime": "2025-04-04T21:19:34.91508"}, {"thingid": "dht1", "temperature": 37, "humidity": 35, "co2": 783, "datetime": "2025-04-04T21:19:36.917201"}, {"thingid": "dht1", "temperature": 40, "humidity": 55, "co2": 1144, "datetime": "2025-04-04T21:19:38.918248"}, {"thingid": "dht1", "temperature": 37, "humidity": 78, "co2": 1054, "datetime": "2025-04-04T21:19:40.920122"}, {"thingid": "dht1", "temperature": 31, "humidity": 64, "co2": 664, "datetime": "2025-04-04T21:19:42.921103"}, {"thingid": "dht1", "temperature": 34, "humidity": 37, "co2": 1177, "datetime": "2025-04-04T21:19:44.921883"}, {"thingid": "dht1", "temperature": 23, "humidity": 54, "co2": 991, "datetime": "2025-04-04T21:19:46.923792"}, {"thingid": "dht1", "temperature": 23, "humidity": 41, "co2": 773, "datetime": "2025-04-04T21:19:48.929281"}, {"thingid": "dht1", "temperature": 32, "humidity": 59, "co2": 987, "datetime": "2025-04-04T21:19:50.931485"}, {"thingid": "dht1", "temperature": 37, "humidity": 54, "co2": 1136, "datetime": "2025-04-04T21:19:53.93599"}, {"thingid": "dht1", "temperature": 39, "humidity": 52, "co2": 917, "datetime": "2025-04-04T21:19:54.93625"}, {"thingid": "dht1", "temperature": 33, "humidity": 45, "co2": 609, "datetime": "2025-04-04T21:19:56.93694"}, {"thingid": "dht1", "temperature": 38, "humidity": 54, "co2": 246, "datetime": "2025-04-04T21:19:58.942515"}, {"thingid": "dht1", "temperature": 34, "humidity": 32, "co2": 1233, "datetime": "2025-04-04T21:20:00.944688"}, {"thingid": "dht1", "temperature": 29, "humidity": 49, "co2": 959, "datetime": "2025-04-04T21:20:02.946736"}, {"thingid": "dht1", "temperature": 28, "humidity": 47, "co2": 828, "datetime": "2025-04-04T21:20:04.949641"}, {"thingid": "dht1", "temperature": 48, "humidity": 55, "co2": 711, "datetime": "2025-04-04T21:20:06.95328"}, {"thingid": "dht1", "temperature": 26, "humidity": 48, "co2": 1132, "datetime": "2025-04-04T21:20:08.955749"}, {"thingid": "dht1", "temperature": 30, "humidity": 59, "co2": 434, "datetime": "2025-04-04T21:20:10.95625"}, {"thingid": "dht1", "temperature": 39, "humidity": 53, "co2": 645, "datetime": "2025-04-04T21:20:12.956867"}, {"thingid": "dht1", "temperature": 34, "humidity": 49, "co2": 645, "datetime": "2025-04-04T21:20:14.978896"}, {"thingid": "dht1", "temperature": 20, "humidity": 69, "co2": 785, "datetime": "2025-04-04T21:20:16.97852"}, {"thingid": "dht1", "temperature": 27, "humidity": 32, "co2": 1252, "datetime": "2025-04-04T21:20:18.97399"}, {"thingid": "dht1", "temperature": 29, "humidity": 35, "co2": 1252, "datetime": "2025-04-04T21:20:20.97265"}, {"thingid": "dht1", "temperature": 25, "humidity": 57, "co2": 1125, "datetime": "2025-04-04T21:20:24.978725"}, {"thingid": "dht1", "temperature": 23, "humidity": 50, "co2": 889, "datetime": "2025-04-04T21:20:26.97952"}, {"thingid": "dht1", "temperature": 24, "humidity": 46, "co2": 1020, "datetime": "2025-04-04T21:20:28.901787"}, {"thingid": "dht1", "temperature": 36, "humidity": 57, "co2": 847, "datetime": "2025-04-04T21:20:30.98756"}, {"thingid": "dht1", "temperature": 39, "humidity": 50, "co2": 1363, "datetime": "2025-04-04T21:20:32.999964"}, {"thingid": "dht1", "temperature": 23, "humidity": 46, "co2": 861, "datetime": "2025-04-04T21:20:34.99618"}, {"thingid": "dht1", "temperature": 40, "humidity": 47, "co2": 873, "datetime": "2025-04-04T21:20:36.997885"}, {"thingid": "dht1", "temperature": 22, "humidity": 35, "co2": 1257, "datetime": "2025-04-04T21:20:38.999155"}, {"thingid": "dht1", "temperature": 35, "humidity": 70, "co2": 761, "datetime": "2025-04-04T21:20:41.980249"}, {"thingid": "dht1", "temperature": 29, "humidity": 31, "co2": 1025, "datetime": "2025-04-04T21:20:43.981305"}, {"thingid": "dht1", "temperature": 27, "humidity": 53, "co2": 1025, "datetime": "2025-04-04T21:20:45.982665"}, {"thingid": "dht1", "temperature": 26, "humidity": 56, "co2": 656, "datetime": "2025-04-04T21:21.93.940319}, {"thingid": "dht1", "temperature": 33, "humidity": 51, "co2": 949, "datetime": "2025-04-04T21:21.95.941765"}, {"thingid": "dht1", "temperature": 37, "humidity": 35, "co2": 1111, "datetime": "2025-04-04T21:24.941344"}, {"thingid": "dht1", "temperature": 21, "humidity": 54, "co2": 992, "datetime": "2025-04-04T21:20:51.925013"}, {"thingid": "dht1", "temperature": 23, "humidity": 56, "co2": 844, "datetime": "2025-04-04T21:20:53.929494"}, {"thingid": "dht1", "temperature": 40, "humidity": 52, "co2": 542, "datetime": "2025-04-04T21:20:55.930769"}, {"thingid": "dht1", "temperature": 26, "humidity": 53, "co2": 918, "datetime": "2025-04-04T21:20:57.937174"}, {"thingid": "dht1", "temperature": 33, "humidity": 47, "co2": 447, "datetime": "2025-04-04T21:20:59.938186"}, {"thingid": "dht1", "temperature": 24, "humidity": 74, "co2": 660, "datetime": "2025-04-04T21:21.01.93919"}, {"thingid": "dht1", "temperature": 28, "humidity": 57, "co2": 656, "datetime": "2025-04-04T21:21.93.940319"}, {"thingid": "dht1", "temperature": 33, "humidity": 51, "co2": 949, "datetime": "2025-04-04T21:21.95.941765"}, {"thingid": "dht1", "temperature": 25, "humidity": 35, "co2": 1223, "datetime": "2025-04-04T21:21.97.945827"}, {"thingid": "dht1", "temperature": 23, "humidity": 55, "co2": 871, "datetime": "2025-04-04T21:21.99.947263"}, {"thingid": "dht1", "temperature": 32, "humidity": 41, "co2": 1118, "datetime": "2025-04-04T21:21.11.948477"}, {"thingid": "dht1", "temperature": 40, "humidity": 76, "co2": 1046, "datetime": "2025-04-04T21:21.13.950003"}, {"thingid": "dht1", "temperature": 37, "humidity": 37, "co2": 1039, "datetime": "2025-04-04T21:21.14.951413"}, {"thingid": "dht1", "temperature": 29, "humidity": 50, "co2": 70, "datetime": "2025-04-04T21:21.17.952174"}, {"thingid": "dht1", "temperature": 40, "humidity": 70, "co2": 405, "datetime": "2025-04-04T21:21.19.953919"}, {"thingid": "dht1", "temperature": 38, "humidity": 44, "co2": 1174, "datetime": "2025-04-04T21:21.21.955238"},
```

Reflection :

This exercise gave me hands-on experience in developing an actual IoT data pipeline using AWS services. It made me understand how various cloud components - AWS IoT Core, Kinesis Firehose, and S3 - can be integrated together to mimic and store sensor data effectively. Key management and security certificate implementation made sure there is firm compliance with the function of secure communication in IoT usage. On top of that, Firehose setup and IAM role creation gave practical insight into data movement and access control management with AWS tools. There were a few technical problems that I faced during this process, especially when I tried to integrate the MQTT simulator with the AWS IoT Core endpoint and configure the Firehose destination. These were dealt with by proper troubleshooting and referring to AWS documentation and examples. Overall, this exercise taught key concepts in cloud-based IoT development and how data can flow seamlessly from mock devices to the cloud storage. Effective end-to-end implementation gave a sense of achievement and a solid grounding for more involved IoT and analytics projects.

GitHub URL : <https://github.com/sivaramakrishna6768/aws-iot-mqtt-simulator-2>