How to Integrate dataFEED OPC Suite Data in AWS IoT
How to Configure Softing *dataFEED* Gateways as AWS IoT Device

**Preliminary Remarks**

This Configuration Manual describes how to configure *dataFEED OPC Suite* as AWS IoT device (“Thing”). From the AWS IoT point of view *dataFEED OPC Suite* is referred to in this manual as *gateway*. It is based on the AWS Developer Guide [https://docs.aws.amazon.com/iot/latest/developerguide/what-is-aws-iot.html](https://docs.aws.amazon.com/iot/latest/developerguide/what-is-aws-iot.html).

Additional information about the Softing product can be found at the following web page

**AWS Preparation Steps**

1. **Register Gateway as Device in the Registry**

Follow the instructions given at [https://docs.aws.amazon.com/iot/latest/developerguide/register-device.html](https://docs.aws.amazon.com/iot/latest/developerguide/register-device.html).

In step 4. define unique *gateway* name, e.g. by using its serial number.

2. **Create and Activate Certificate for Gateway**

Follow the instructions given at [https://docs.aws.amazon.com/iot/latest/developerguide/create-device-certificate.html](https://docs.aws.amazon.com/iot/latest/developerguide/create-device-certificate.html).

In step 2. download certificate and key files. These files need to be uploaded as MQTT Client certificate into the *gateway* in a later step.
3. **Create AWS IoT Policy for MQTT Client Certificate**

Follow the instructions given at [https://docs.aws.amazon.com/iot/latest/developerguide/create-iot-policy.html](https://docs.aws.amazon.com/iot/latest/developerguide/create-iot-policy.html).

Define **gateway** permission for **iot:Connect** and **iot:Publish** actions.

**NOTE:**
For some unknown reasons the attachment of policies as described in step 3. does not work. Instead, it looks like the policy must be created before attaching it to a certificate. However, the certificate dialog does not trigger the creation of a policy.

It is possible to create and attach a policy after the creation of a certificate by following these steps:

- Return to **AWS IoT Console** main screen and navigate to **Secure/Policies**

  ![AWS IoT Console](image)

  - Press **Create a policy** button
  - Define policy name
  - Define **gateway** permission for **iot:Connect** action
- Enter user-defined client ID in **Resource ARN** field
  It is recommended to use the **gateway** name here. Remember this client ID for a later step.

- Press **Add Statement** button
- Define **gateway** permission for **iot:Publish** action
- Define user-specific ARN topic. Remember this topic for a later step.

**NOTE:**
The placeholder “*” allows publishing to any topic.
• **Press *Create* button**

<table>
<thead>
<tr>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>uagate-163300167-policy</td>
</tr>
</tbody>
</table>

  **Add statements**
  
  Policy statements define the types of actions that can be performed by a resource.

<table>
<thead>
<tr>
<th>Action</th>
<th>Resource ARN</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow</td>
</tr>
<tr>
<td>Deny</td>
</tr>
</tbody>
</table>

  **Add statement**

  4. **Attach AWS IoT Policy to Gateway**

  • Return to *AWS IoT Console* main screen and navigate to *Manage/Things* page
  • Click on device name of *gateway*
• Click on **Security**
• Click on hash value of created device certificate

![Image of Security section with hash value highlighted]

• Select **Attach policy** in **Actions** menu

![Image of Attach policy highlighted]

• Attach created policy to certificate

**Attach policies to certificate(s)**

Policies will be attached to the following certificate(s):
2c1696c35411d66fae3efb608ef929e15363f3773088f07d0bbd47cd45e325e

Choose one or more policies

- **uagate-163300167-policy**

1 policy selected  
Cancel  
Attach
5. Determine AWS MQTT Broker Address
   • Return to AWS IoT Console main screen and navigate to Settings
   • MQTT Broker address can be found in Custom Endpoint field
   **NOTE:**
   The address looks similar to “123456789-ats.iot.eu-central-1.amazonaws.com”

**dataFEED OPC Suite Configuration Steps**

6. Define MQTT Broker Connection
   • Start dataFEED OPC Suite Configurator
   • Navigate to Data Destination/MQTT Broker page
• Navigate to **Connection Settings** page (page 1) of **MQTT Broker Connection Wizard**

- **Connection Settings**
  - **Connection Name**
    - Provide here the connection name which will identify the current connection. The name must be unique throughout the whole configuration.
    - **Connection Name**: MQTT.Broker.dev

- **Client ID**
  - **Client ID** field: dataPubSub

- **Connection State**
  - **Connection Active**: Yes

**NOTE:**
Device ID has to be allowed in iot:Connect policy (see section **AWS Preparation Steps**)
- Navigate to **Communication Settings** page (page 2) of **MQTT Broker Connection Wizard**

  ![MQTT Broker Connection Wizard](image)

  **Communication Settings**

  On this wizard page, the communication settings of the data destination connection to an external MQTT broker for publishing data are configured.

  **MQTT Broker Settings**

  **MQTT Broker URI**

  - `<IP address or hostname with domain of the broker> [: <port number>]`

  - **ssl://**

  - Enter **AWS MQTT broker address** as prefix in `<IP address or hostname with domain of the broker> [: <port number>]` field

  - Enter port number “**8883**” as postfix in `<IP address or hostname with domain of the broker> [: <port number>]` field

  - Enable **MQTT Clean Session** flag

  **NOTE:**

  AWS Broker disconnects, if flag is not set

  (see [https://docs.aws.amazon.com/iot/latest/developerguide/protocols.html](https://docs.aws.amazon.com/iot/latest/developerguide/protocols.html))

- Set **Authentication Settings/User Identity** to **Anonymous**

- Activate **Use Client Certificate** checkbox

- Select the downloaded certificate “*-certificate.pem” and private key “*-private.pem.key”

  **NOTE:**

  Before the private key can be selected, the certificate file extension has to be changed to “.pem”
MQTT Topic definition

On this wizard page the address space for the current connection can be defined.

Connection: MQTT-Broker-axx

MQTT Topics

Add the topic you want to insert into the namespace of the local application. The topics can be defined manually or imported from an external file. Please define the item mapping into MQTT topics within the configuration of the Exchange function. The configuration of the Exchange function can be found within Data Processing.

MQTT-Broker-axx

Topic 1

- Time Zone Format

Specify the time zone for the timestamp expression used within requests.

Time zone: UTC
• Navigate to **MQTT Topic definition** page (page 3) of **MQTT Broker Connection Wizard**

![MQTT Topic definition page](image)

• Add topics

  **NOTE:**
  Topics have to be allowed in iot:Publish policy
  (see section **AWS Preparation Steps**)

• Deactivate **Retain** flag

• Press **Save** button
• Navigate to **Data Processing/Exchange** page

![Configuration Manual](image)

7. **Finish dataFEED OPC Suite Configuration**
   • **dataFEED OPC Suite** configuration is finished
   • Save configuration and start **dataFEED OPC Suite**
8. **Monitor Values in AWS**

- Return to *AWS IoT Console* main screen and navigate to *Test*
- Subscribe to all topics using the wildcard “#”
- Received MQTT messages look different, depending on topic format settings and options for the collected PLC values

### MQTT client for single value

![Image of MQTT client for single value]

### MQTT client for multi value

![Image of MQTT client for multi value]