7310-8 LoRaWAN Gateway with AWS IoT Core

Getting Started Guide

7310-8 LoRaWAN Gateway with AWS IoT Core

Getting Started Guide

Table of Contents

1. Document Information

1.1 Naming Conventions

The terms “downlink device” or “endpoint device” are used in this document to refer to a LoRaWAN device that connects to a LoRaWAN “gateway.” The term “gateway,” in turn, refers to the LoRaWAN device that connects to the AWS IoT Core for LoRaWAN.

1.2 Revision History

Revision A – Initial release
Release Date – March 31, 2021

2. Overview

The ADTRAN 7310-8 is a micro-sized indoor Long Range Wide Area Network (LoRaWAN) Internet of Things (IoT) gateway that is perfect for enterprises, small businesses, and industrial IoT applications. This 8-channel gateway supports an external antenna, one Gigabit interface, and can be powered by an 802.11af PoE or Micro-USB connector. This device provides a reliable connection to a public or private LoRaWAN. It is designed and built specifically for IoT machine-to-machine communications.

3. Hardware Description

3.1 Data Sheet

The data sheet for the ADTRAN 7310-8 gateway can be found here.

3.2 Standard Kit Contents

The ADTRAN 7310-8 gateway includes the following items in the standard kit:

- ADTRAN’s 8-channel Enterprise LoRaWAN gateway
- One external antenna
- One RJ-45 Ethernet cable
- One Micro-USB cable and wall charger
- One mounting kit containing:
    - Wall/Ceiling mount hardware: mounting plate, two anchors, and associated screws
    - Two T-rail mounting brackets (9/16 and 15/16-inch) for recessed drop ceiling
- One quick start guide

3.3 User Provided items

Users must provide their own LoRaWAN end devices and an Apple smartphone or other device (to use the iOS ADTRAN IoT app).

3.4 3rd Party purchasable items

None.

3.5 Additional Hardware References

Additional hardware information can be found in the ADTRAN 7310-8 Quick Start Guide.

4. Configuring your AWS Account and Permissions

If you have not already created an AWS account, refer to the instructions provided in the AWS IoT Developer’s Guide, available online here. The relevant sections of that guide for account creation are the ***Sign up for an AWS account*** and ***Create a user and grant permissions*** sections.

Once you have created an AWS account and specified your permissions, you can proceed with the getting started information included in the following sections of this guide to configure and use your ADTRAN 7310-8 gateway with the AWS IoT Core for LoRaWAN.

5. Getting Started with ADTRAN's 7310-8 Gateway

The following sections outline the necessary steps to configure and register the ADTRAN 7310-8 gateway and any additional endpoint devices with the AWS IoT Core for LoRaWAN.

5.1 Getting Started Configuration Overview

To get started with the AWS IoT Core for LoRaWAN, you will need to configure the IAM roles and policies for the Configuration and Update Server (CUPS) and AWS IoT destination role, register the LoRaWAN gateway with the AWS IoT Core, and add any LoRaWAN endpoint devices to the AWS IoT Core.

The following sections describe the steps necessary to configure the 7310-8 gateway and additional endpoint devices for use with the AWS IoT Core for LoRaWAN:

- Configuring Roles and Policies in IAM
- Registering the LoRaWAN Gateway with AWS IoT
- Adding a LoRaWAN Endpoint Device to AWS IoT

***NOTE**: The examples in this document are intended only for dev environments. All devices in your fleet must have credentials with privileges that authorize only intended actions on specific resources. The specific permission policies can vary for your use case. Identify the permission policies that best meet your business and security requirements. For more information, refer to Example policies and Security Best practices.*

5.2 Configuring Roles and Policies in IAM

Two steps are required for configuring roles and policies in IAM. First, you must create an IAM role for the Configuration and Update Server (CUPS) and review its associated policies, and then you must create an AWS IoT Core destination role and review its associated policies. The steps required to create these roles and policies are detailed in the following sections.

5.2.1 Adding an IAM Role for CUPS server

The first role to be configured for the AWS account is the IAM role for the CUPS server. This role allows the server to handle the wireless gateway credentials. This procedure must be performed before a LoRaWAN gateway attempts to connect with the AWS IoT Core for LoRaWAN, but only needs to be completed once.

To configure the IAM role for the CUPS server, connect to the IAM Roles page on the IAM console and follow these steps:

1. From the IAM Roles page on the IAM console select **Create Role**.
2. In the **Create Role** menu, under **Select type of trusted entity**, select **Another AWS Account**.
3. Enter your account ID in the **Account ID** field and select **Next: Permissions**.
4. In the **Permissions** menu, enter *AWSIoTWirelessGatewayCertManager* in the **Filter Policies** search field and select search. If the search results display the policy named *AWSIoTWirelessGatewayCertManager*, select the check box next to that policy name and proceed to Step 5 to create the role.

   If the search results do not include the *AWSIoTWirelessGatewayCertManager* policy, you must create the policy before creating the role. Create the policy by following these steps:

   a. Return to the IAM console and select **Policies** from the menu on the left.

   b. In the **Policies** menu, select **Create Policy** and then the **JSON** tab. Selecting the **JSON** tab will open the policy editor where you will replace the existing policy template with the following trust policy information:

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
                {
                    "Sid": "IoTWirelessGatewayCertManager",
                    "Effect": "Allow",
                    "Action": [
                        "iot:CreateKeysAndCertificate",
                        "iot:DescribeCertificate",
                        "iot:ListCertificates",
                        "iot:RegisterCertificate"
                    ],
                    "Resource": "*"
                }
            ]
        }
```

c. After updating the policy, select **Review Policy** to open the **Review Policy** page and make the following changes:
   - In the **Name** field, enter *AWSIoTWirelessGatewayCertManager*.

     NOTE: *You **MUST** enter the policy name as **AWSIoTWirelessGatewayCertManager** and cannot use a different name. This is for consistency with future releases.*
   - In the **Description** field, enter a description of your choice.

d. After reviewing the policy and specifying the name and description, select **Create Policy** to create the policy. A confirmation message indicating that the policy has been created is displayed.

5. Once the correct policy is specified, select **Next: Tags** and then **Next: Review** to review and create the IAM role.
6. In the role review page, enter *IoTWirelessGatewayCertManagerRole* in the **Role Name** field and select **Create Role** to the create the IAM role.

   NOTE: *You **MUST** enter the role name as **IoTWirelessGatewayCertManagerRole** and cannot use a different name. This is for consistency with future releases.*
7. Once the role is created, you will need to specify the trust relationships and policies for the role. In the confirmation message indicating that the role has been created, select **IoTWirelessGatewayCertManagerRole** to edit the newly created role.
8. In the resulting role **Summary** page, select the **Trust Relationships** tab and then select **Edit Trust Relationship**.
9. Navigate to the **Policy Document** for the role's trust relationship and change the **Principal** property to represent the IoT wireless service. The **Principal** value should look like the following:

```
"Principal": {
    "Service": "iotwireless.amazonaws.com"
}
```

10. After changing the **Principal** property, the complete policy document should look like the following:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
        "Effect": "Allow",
        "Principal": {
          "Service": "iotwireless.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {}
    }
  ]
}
```

11. If the policy is specified correctly, select **Update Trust Policy** to save your changes and exit the role configuration.

Once these steps are completed, you have successfully configured the IAM CUPS role (*IoTWirelessGatewayCertManagerRole*) and can proceed to configuring the AWS IoT Core destination role.

5.2.2 Adding an IAM Role for the AWS IoT Core for LoRaWAN Destination

The second role to be configured for the AWS account is AWS IoT Core destination role. This role is allows your AWS account to operate with the AWS IoT Core for LoRaWAN and is configured by first defining the policy associated with the role, and then creating the role itself.

To create a policy that gives the role permission to describe the IoT endpoint and publish messages to AWS IoT Core, follow these steps:

1. Connect to the IAM console and select **Policies** from the menu on the left.
2. In the **Policies** menu, select **Create Policy** and then the **JSON** tab. Selecting the **JSON** tab will open the policy editor where you will replace the existing policy template with the following trust policy information:

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action":
         [
           "iot:DescribeEndpoint",
           "iot:Publish"
         ],
        "Resource": "*"
      }
    ]
}
```

3. After updating the policy, select **Review Policy** to open the **Review Policy** page and specify a policy name of your choice in the **Name** field and a description of your choice in the **Description** field.
4. After reviewing the policy and specifying the name and description, select **Create Policy** to create the policy. A confirmation message indicating that the policy has been created is displayed.

Once the policy for the destination role has been successfully created, you can begin configuring the destination role itself. To create the destination role, connect to the IAM console and follow these steps:

1. Select **Roles** from the menu on the left and then select **Create Role**.
2. In the **Create Role** menu, under **Select type of trusted entity**, select **Another AWS Account**.
3. Enter your account ID in the **Account ID** field and select **Next: Permissions**.
4. In the **Permissions** menu, enter the name of the policy you just created for the destination role in the **Filter Policies** search field and select search. Select the check box next to the appropriate policy name to begin configuring role to which this policy will be applied.
5. Once the correct policy is selected from the list, select **Next: Tags** and then **Next: Review** to review the role's configuration settings.
6. In the role review page, enter a role name of your choice in the **Role Name** field and a description of your choice in the **Description** field and select **Create Role** to the create the IAM destination role.
7. Once the role is created, you will need to specify the trust relationships and policies for the role to grant the AWS IoT Core for LoRaWAN permission to assume this IAM role when delivering messages from devices to your AWS account. In the confirmation message that indicates the role has been created, select the name you specified for this role to edit the role.
8. In the resulting role **Summary** page, select the **Trust Relationships** tab and then select **Edit Trust Relationship**. The principal AWS role in your trust policy document defaults to root and must be changed.
9. To change the principal AWS role in the trust policy document, navigate to the **Policy Document** for the role's trust relationship and replace the existing policy with the following:

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
```

```
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "iotwireless.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {}
        }
    ]
}
```

10. Select **Update Trust Policy** to apply the policy changes. Once the changes are applied, in the **Trusted Entities** section you will see the following message: *The identity provider(s) iotwireless.amazonaws.com*.

Once the destination role has been created and configured to communicate with the AWS IoT Core for LoRaWAN, the IAM role and policy configuration is complete. The next step in getting started with the AWS IoT Core is to register the LoRaWAN device with the AWS IoT Core.

5.3 Registering the LoRaWAN Gateway with AWS IoT

Once the IAM roles and policies are configured for your AWS account, the next step in getting started with the AWS IoT Core is to register the LoRaWAN gateway with the AWS IoT Core. To successfully register the gateway, gather the required information for the gateway and complete the steps detailed in Registering the LoRaWAN Gateway below.

5.3.1 Preparation

To complete the gateway registration process, you will need the following information available:

- **LoRaWAN region**. For example, if the gateway is deployed in a US region, the gateway must support LoRaWAN region US915.
- **Gateway LNS-protocols**. Currently, the LoRa Basics Station 2.0.5 protocol is supported.
- **Gateway ID (GatewayEUI) or serial number**. This is used to establish the connection between the LNS and the gateway. Consult the documentation for your gateway to locate this value.

5.3.2 Registering the LoRaWAN Gateway

To register the LoRaWAN gateway with AWS IoT Core for LoRaWAN, connect to the AWS IoT console and follow these steps:

1. Select **Wireless Connectivity** from the menu on the left.
2. In the **Wireless Connectivity** menu, select **Intro** and then select **Get Started**. This step is needed to pre-populate the default profiles for the device.
3. In the **Get Started** menu, navigate to the **Add LoRaWAN Gateways and Wireless Devices** section and select **Add Gateway**.
4. In the **Add Gateway** menu, enter the appropriate information in the **GatewayEUI** and **Frequency Band (RF Region)** fields. The Gateway EUI value is the device's MAC address with **FFEE** inserted between the third and fourth octets. For example, for a gateway with MAC address **XX:XX:XX:XX**, the EUI value would be **XX:XX:FF:EE:XX:XX** .

   NOTE: *The ADTRAN 7310-8 gateway's MAC address can be found in the* ***Ethernet Config*** *menu within the iOS ADTRAN IoT app. Specific instructions for accessing the app and locating this menu are detailed in Configuring the LoRaWAN Gateway (Basic Configuration) and ADTRAN 7310-8 Gateway Software Configuration.*
5. Next, enter a descriptive name in the **Name – optional** field (ADTRAN recommends that you use the GatewayEUI as the name), and select **Add Gateway**.

Once the gateway has been successfully added, you will need to specify the certificate information for the gateway. To specify the certificate and complete the registration process, follow these steps:

1. After selecting **Add Gateway** to add the gateway to the device list, from the **Get Started** menu, select **Configure Your Gateway** to access the gateway configuration menu.

2. In the **Configure Your Gateway** menu, navigate to the **Gateway Certificate** section and select **Create Certificate**.
3. Once the **Certificate created and associated with your gateway** message is displayed, select **Download Certificates** to download the certificate (**xxxxx.cert.pem**) and the associated private key (**xxxxxx.private.key**).
4. After the certificates and private key have been downloaded, navigate to the **Provisioning Credentials** section of the **Configure Your Gateway** menu and select **Download Server Trust Certificates**. Selecting this option will download the CUPS (**cups.trust**) and LNS (**lns.trust**) server trust certificates. Be sure to copy and save the provided CUPS and LNS endpoint information; it will be required when configuring the gateway for use (as described in Configuring the ADTRAN 7310-8 Gateway for AWS).
5. After downloading the certificates, private key, and server trust certificates, select **Submit** to complete the gateway registration process with the AWS IoT Core for LoRaWAN.

5.4 Adding a LoRaWAN Endpoint Device to AWS IoT

Once your gateway has been registered with the AWS IoT Core, you can begin registering LoRaWAN endpoint devices to the AWS IoT Core. The registration process for each LoRaWAN endpoint device includes verifying the device and service profiles for the device, configuring a destination for the device, and finally, registering the device. To successfully register the device and add it to the AWS IoT Core, gather the required information for the device and complete the steps detailed in the following sections.

5.4.1 Preparation

To complete the endpoint device registration process, you will need the following information available:

- **LoRaWAN region**. This must match the gateway LoRaWAN region (as specified in 5.3.2 Registering the LoRaWAN Gateway). The following frequency bands (RF regions) are supported:
  - EU868
  - US915
  - EU433
- **MAC version**. The following MAC versions are supported:
  - V1.0.2
  - v1.0.3
  - v1.1
- **OTAA version**. OTAA v1.0x and OTAA v1.1 are supported.
- **ABP version**. ABP v1.0x and ABP v1.1 are supported.

In addition, you will need to locate and note the following information from your endpoint device manufacturer:

- For OTAA v1.0x devices: DevEUI, AppKey, AppEUI
- For OTAA v1.1 devices: DevEUI, AppKey, NwkKey, JoinEUI
- For ABP v1.0x devices: DevEUI, DevAddr, NwkSkey, AppSkey
- For ABP v1.1 devices: DevEUI, DevAddr, NwkSEnckey, FNwkSIntKey, SNwkSIntKey, AppSKey

5.4.2 Verifying Endpoint Device and Service Profiles

AWS IoT Core for LoRaWAN supports endpoint device profiles and service profiles; verifying these profiles for your endpoint device is the first step in device registration. Device profiles contain the communication and protocol parameter values the device needs to communicate with the network server, whereas service profiles describe the communication parameters the device needs to communicate with the application server.

NOTE: *Some pre-defined profiles are available for device and service profiles. Before proceeding, you must verify that these default profile settings match the devices you will be configuring to work with AWS IoT Core for LoRaWAN.*

To view the available default device profiles, verify that these profiles will work for your device, and optionally configure the device profile parameters, connect to the AWS IoT console and follow these steps:

1. In the navigation pane, select **Wireless connectivity** and then select **Profiles**.
2. In the **Profiles** menu, navigate to the **Device Profiles** section and view the pre-defined device profiles listed. Check each of the profiles to determine if one of them will work for your device.
3. If no default device profiles are suitable, you can define your own device profile by selecting **Add device profile** and specifying the parameters as needed. For example, if you are adding a device profile for US 915, the values you would specify are:
    - MacVersion 1.0.3
    - RegParamsRevision RP002-1.0.1
    - MaxEirp 10
    - MaxDutyCycle 10
    - RfRegion US915
    - SupportsJoin true

Once you have a device profile specified that will work for you, you can continue profile verification by reviewing the available service profiles. To verify the service profiles associated with this device, follow these steps:

4. In the **Service Profiles** section of the **Profiles** menu, view the pre-defined service profiles listed. Check each of the profiles to determine if one of them will work for your device.
5. If no default service profiles are suitable, you can define your own service profile by selecting **Add service profile** and specifying the parameters as needed. For your reference, the default service profile parameters are shown below. Note however, that only the **AddGwMetadata** parameter can be changed at this time.
    - UlRate60
    - UlBucketSize4096
    - DlRate60
    - DlBucketSize4096
    - AddGwMetadatatrue
    - DevStatusReqFreq 24
    - DrMax15
    - TargetPer5
    - MinGwDiversity1

Proceed in the device configuration only if you have a device and service profile that will work for you.

5.4.3 Configuring a Destination for Endpoint Device Traffic

Once the device and service profiles for your endpoint device have been determined, the next step in device registration is to specify a destination for endpoint device traffic. Because most LoRaWAN devices do not send data to AWS IoT Core for LoRaWAN in a format that can be consumed by AWS services, traffic must first be sent to a configured destination. A configured destination represents the AWS IoT rule that processes a device's data for use by AWS services. This AWS IoT rule contains the SQL statement that selects the device's data and the topic rule actions that send the result of the SQL statement to the services that will use it.

NOTE: *For more information on destinations, including their configuration and use, refer to the AWS LoRaWAN developer guide.*

Destination configuration consists of specifying a role and a rule for the destination. The role associated with the destination is the same IAM role configured in Adding an IAM Role for the AWS IoT Core for LoRaWAN Destination. To associate the previously configured destination role with this destination, and specify a destination rule, connect to the AWS IoT console and follow these steps:

1. In the navigation pane, select **Wireless connectivity**, followed by **Destinations**, and then finally select **Add Destination**.
2. On the **Add destination** page, in the **Permissions** section, select the previously created IAM role from the drop-down menu.

3. Under **Destination details**, enter *ProcessLoRa* as the **Destination name**, and an appropriate description in the **Destination description – optional** field.

   ***NOTE:** The **Destination name** can be anything. For getting started and consistency, choose **ProcessLoRa** for the first integration with AWS IoT Core for LoRaWAN.*

4. Once the destination role is specified, begin configuring the destination rule by entering *LoRaWANRouting* in the **Rule name** field. Ignore the **Rules configuration – Optional** section for now. The destination rule will be fully configured later, as described in the "Hello World" sample application (refer to Creating the Destination Rule in the "Hello World" configuration section).
5. Once you have specified the destination role and rule name, select **Add Destination**. The "*Destination added*" message will display, indicating that the destination has been successfully added.

5.4.4 Registering the Endpoint Device

Once the endpoint device destination has been specified, you can begin the endpoint device registration process with the AWS IoT Core for LoRaWAN by connecting to the AWS IoT console and following these steps:

1. In the navigation pane, select **Wireless connectivity**, followed by **Devices**, and then finally select **Add wireless device**.
2. In the **Add device** page, select the appropriate LoRaWAN specification version from the **Wireless device specification** drop-down menu.
3. In the **LoRaWAN specification and wireless device configuration** menu, enter the **DevEUI** value in the appropriate field, and again in the **Confirm DevEUI** field to confirm it.
4. Complete the remaining fields as per the OTAA/ABP choice for your device (as defined in 5.4.1 Preparation). Be sure to enter a name for your device in the **Wireless device name – optional** field.
5. Navigate to the **Profiles** section (under **Wireless device profile**), select the appropriate profile from the drop-down menu, and select **Next**. The selected profile should correspond to your device and region.

   ***NOTE:** Compare your device details to the profile to ensure the device profile is correct. If there are no valid default options, you will have to create a new profile (as detailed in 5.4.2 Verifying Endpoint Device and Service Profiles).*
6. Next, select the previously created destination for the endpoint device (*ProcessLoRa*) from the **Choose destination** drop-down menu and select **Add device**. A "*Wireless device added*" message will display, indicating that your endpoint device has been successfully registered to the AWS IoT Core.

6. Configuring the ADTRAN 7310-8 LoRaWAN Gateway

The following sections detail the processes for installing the ADTRAN 7310-8 LoRaWAN gateway hardware and software. Additional information for both processes is available online in the ADTRAN 7310-8 Quick Start Guide.

6.1 ADTRAN 7310-8 Hardware Installation

The hardware installation process for the ADTRAN 7310-8 gateway includes the following:

- Installing the Antenna
- Mounting the 7310-8 LoRaWAN Gateway
- Supplying Power to the 7310-8 LoRaWAN Gateway
- Connecting to the LoRaWAN Gateway
- Configuring the LoRaWAN Gateway
- Understanding the 7310-8 Status LEDs

6.1.1 Installing the Antenna

To install the provided antenna onto the LoRaWAN gateway, attach the external antenna onto the antenna port on the left side of the gateway as shown in the ADTRAN 7310-8 Quick Start Guide.

6.1.2 Mounting the 7310-8 LoRaWAN Gateway

The 7310-8 LoRaWAN gateway can be mounted to a ceiling, wall, or dropped ceiling. To mount the gateway in one of these configurations, follow the instructions provided in the ADTRAN 7310-8 Quick Start Guide. When mounting

the gateway, ensure the device is positioned for maximum coverage and range between other gateways and wireless client devices.

6.1.3 Supplying Power to the 7310-8 LoRaWAN Gateway

The 7310-8 LoRaWAN gateway does not have a power switch. It is powered on when connected to a network device that supplies Power over Ethernet (PoE) based on the IEEE 802.3at standard. To power the LoRaWAN gateway, follow these steps:

1. Connect one end of the provided RJ-45 Ethernet cable to the **POE/NET** port on the LoRaWAN gateway.
2. Connect the other end to a PoE-powering network device.
3. Confirm that the power is connected properly. The **POWER** LED should be **ON**, as detailed in Understanding the 7310-8 Status LEDs.

NOTE: *If a PoE-powering device is not available, the micro universal serial bus (micro-USB) connection (labeled* ***Aux/PWR****) can be used for powering the LoRaWAN gateway instead.*

6.1.4 Connecting to the LoRaWAN Gateway

Once the LoRaWAN gateway is installed and powered, you can connect to the LoRaWAN gateway using the Bluetooth ADTRAN IoT app from a supported Apple® device. To connect using the ADTRAN IoT app, follow these steps:

1. Download the ADTRAN IoT app onto your Apple device. The ADTRAN IoT app is available for download from Apple's App Store.
2. Ensure that the Bluetooth feature of your Apple device is on.
3. Open the ADTRAN IoT app on your Apple device and select **SCAN** from the menu. The application will scan for any ADTRAN LoRaWAN gateways within the Bluetooth connection range and provide you the option to connect to any discovered devices. Discovered devices will be displayed; the default name of any discovered device is set to the last 5 digits of the device's serial number (for example, **0007**).
4. To connect to a discovered device, select the device from the list and enter your PIN when prompted. The PIN will always be the last 5 digits of the of the gateway's serial number. After entering the PIN, select **Connect**. Once the connection is complete, the device information and configuration are accessible from the **Connected Device** menu.
5. Once you have connected to the LoRaWAN gateway using the ADTRAN IoT app, you can access the configuration, device information, power settings, and device status information from the ADTRAN IoT app directly. The following configuration menus are available from the ADTRAN IoT app:
    - Gateway Info Menu
    - Gateway Status Menu
    - Ethernet Config Menu
    - LoRa SubBand Menu
    - LoRa Config Menu
    - System Status Menu

6.1.5 Configuring the LoRaWAN Gateway (Basic Configuration)

After you have connected to the 7310-8 LoRaWAN gateway via the ADTRAN IoT app, you can use the app the configure the specific parameters of the gateway. To complete the minimal gateway configuration, use the ADTRAN IoT app to complete the following tasks:

1. The first step in configuring the LoRaWAN gateway is to rename the device so that the last 5 digits of the serial number (the default name of the device and the device PIN) are not distributed. Be sure to note your device PIN before changing the default name of the device, as the PIN will not change but will no longer be displayed in the device name. To change the name of the device, select the **Gateway Info** tab from the **Connected Device** menu. In the **Gateway Info** menu, enter a new device name in the **Device Name** field, and select **Save**.
2. After a successful save, disconnect from the device. The old device name will still be displayed in the available devices list; to update the device name in the device list, log back into the same device using the procedure outlined in **Step 4** of Connecting to the LoRaWAN Gateway, and then disconnect again. The

new device name will be displayed in the **Connected Device** menu (note that it can sometimes take a few moments for the ADTRAN IoT app to reflect the name change).

3. If any additional device configuration is required, make the necessary changes within the ADTRAN IoT app and select **Save** at the top right corner of the device screen where applicable. If the **Save** option does not appear, the changes are automatically pushed to the LoRaWAN gateway.

   NOTE: *Additional configuration information is specified in Configuring the ADTRAN 7310-8 Gateway for AWS.*

6.1.6 Understanding the 7310-8 Status LEDs

The LEDs on the 7310-8 LoRaWAN gateway's front panel provide you with the ability to monitor the device status. The following section describes the five types of LEDs available on the LoRaWAN gateway device.

POWER Status LED

The POWER Status LED indicates if the device is powered up correctly.

LED	Color	State	Description
POWER	Green	Off	The device is not receiving power.
		On	The device is powered ON correctly.

NET TX/RX Status LEDs

The NET TX and NET RX Status LEDs indicate the transmitted and received encapsulated LoRaWAN packets on the port.

LED	Color	State	Description
NET **TXNET** **RX**	Green	Off	Indicates no traffic is being passed on the port.
		Flashing	Indicates transmitted (TX) or received (RX) encapsulated LoRaWAN packets are being passed on the port.

LoRa Status LED

The LoRa Status LED indicates the status of the LoRa radio.

LED	Color	State	Description
LoRa	Green	Off	The LoRa radio is inactive.
		Flashing	The LoRa radio is passing traffic.

LoRaWAN Status LED

The (LoRaWAN Network Server) LNS Status LED indicates the status of the connection between the LoRa gateway and the LNS.

LED	Color	State	Description
LoRaWAN	Green	Off	The LoRaWAN gateway is not connected to the LNS.
		On	The LoRaWAN gateway is connected to the LNS.
		Flashing	The LoRaWAN gateway is attempting to connect to the LNS, but cannot complete the connection.

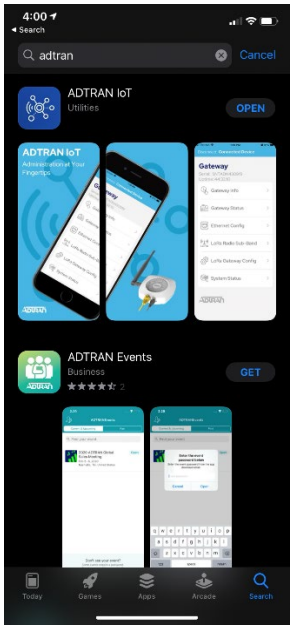
6.2 ADTRAN 7310-8 Gateway Software Configuration

The following sections provide information for downloading the ADTRAN IoT app and using it to connect to the 7310-8 gateway, verify the gateway information, settings, and functions, and to factory reset or reboot the device.

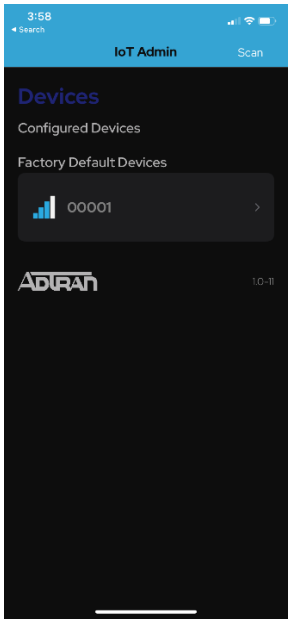
6.2.1 Connecting to the Gateway Using the ADTRAN IoT App

To access the 7310-8 gateway using the iOS-based ADTRAN IoT app, follow these steps:

1. On your iOS device go to the **App Store**.
2. Search for **adtran** and download the **ADTRAN IoT** app.

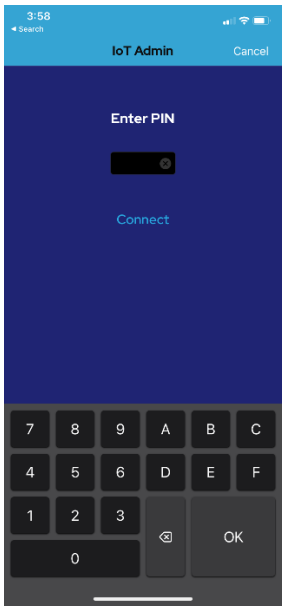


3. Open the **ADTRAN IoT** app and select **Scan** to find your nearby gateways.

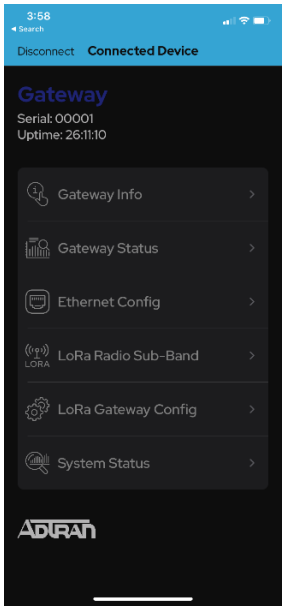


4. Select your gateway from the list by pressing it, entering your unique 5-digit PIN, and selecting **Connect**.

   NOTE: *The PIN is last 5 digits of the device serial number and also the default gateway name. For security, it is suggested to change the gateway name after provisioning. Instructions are outlined in* [*Configuring the LoRaWAN Gateway (Basic Configuration)*]*.*



5. Once the connection is complete, the **Connected Device** screen is displayed, and all configuration menus are available.

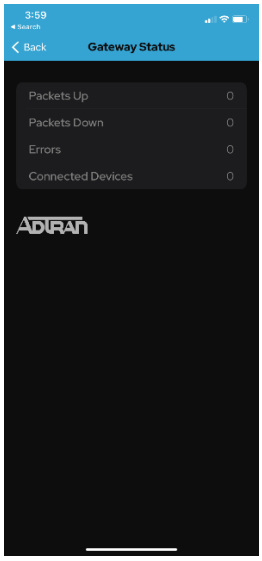


6.2.2 Verifying Gateway Information

You can use the ADTRAN IoT app to verify the 7310-8 gateway information: including hardware and software versions, serial number, part numbers and model information, as well as traffic statistics, errors, and connected devices.

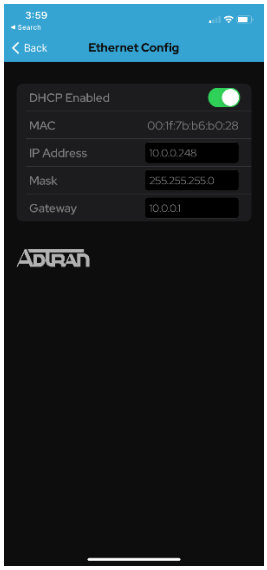
To verify the gateway **Software/Hardware Version**, **Serial Number**, **Adtran Part Number**, and **Model**, open the ADTRAN IoT app **Connected Device** menu and select **Gateway Info**. The relevant information is listed in this menu.

To view information regarding **Packets Up/Down**, **Errors**, and **Connected Devices**, open the ADTRAN IoT app **Connected Devices** menu and select **Gateway Status**. The relevant information is listed in this menu.



NOTE: *The* ***Connected Devices*** *count is not currently integrated with AWS Basic Station.*

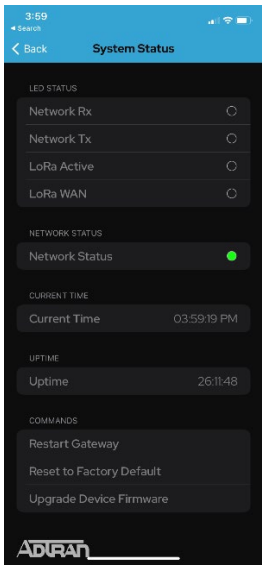
To verify and configure gateway Ethernet settings (such as **DHCP**, **MAC**, **IP Address**, **Mask**, and **Gateway** settings), open the ADTRAN IoT app **Connected Devices** menu and select **Ethernet Config**. The relevant information is listed in this menu.



6.2.3 Restoring the Gateway to Factory Defaults

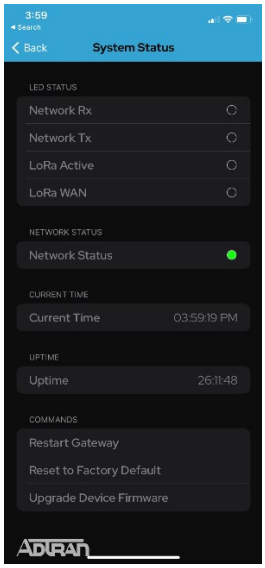
To reset the ADRAN 7310-8 gateway to the factory default settings, connect to the gateway using the ADTRAN IoT app and navigate to the **System Status** menu. Select **Reset to Factory Default** to return the device to the factory default settings.

NOTE: *Restoring the gateway to the factory default settings will clear existing certificates, reset counters, and restore the device name to the last 5 digits of the device serial number.*



6.2.4 Rebooting the Gateway Remotely

To reboot the ADTRAN 7310-8 gateway remotely, connect to the ADTRAN IoT app and navigate to the **System Status** menu. Select **Restart Gateway** to reboot the device.



6.3 Additional Software References

Please browse the ADTRAN Support Community for available technical documentation.

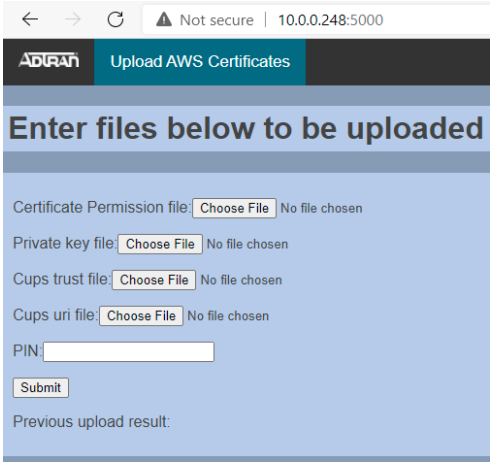
6.4 Configuring the ADTRAN 7310-8 Gateway for AWS

To configure the ADTRAN 7310-8 gateway for use with AWS, you will need to upload the CUPS certifications to the gateway and then provision and start the gateway's packet forwarding program for communication with AWS. The steps required to complete these configurations are provided in the following sections.

6.4.1 Uploading CUPS Certifications to the 7310-8 Gateway

The first step in provisioning the ADTRAN 7310-8 gateway for use with AWS is to upload the configured CUPS certifications to the gateway. To upload the CUPS certificates, follow these steps:

1. Acquire the 7310-8 gateway IP address through either the ADTRAN IoT app or another process.
2. Enter the gateway IP address in a web browser, and be sure to specify port 5000 is used. Enter the IP address and port in the following format: **X.X.X.X:xxxx**; for example, **10.0.0.248:5000**.

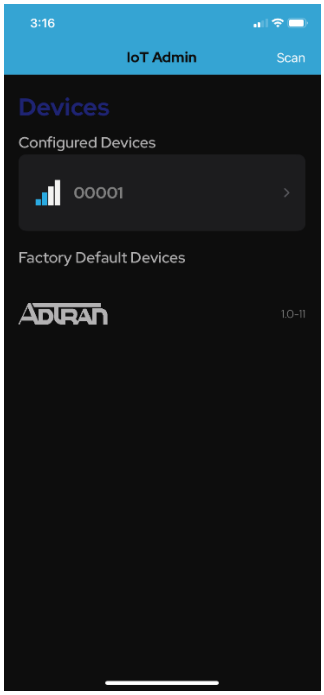


3. Upload the 4 CUPS certificates created in Adding an IAM Role for CUPS Server.
4. Enter the gateway's 5-digit PIN (last 5 digits of the device's serial number).
5. Select **Submit** to upload the certificates to the gateway.

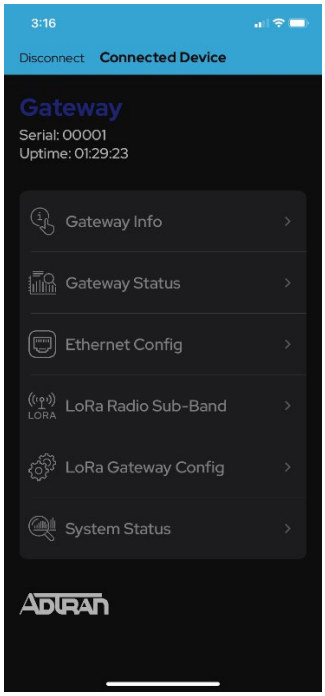
6.4.2 Provisioning and Starting the Packet Forwarding Program

The second step in configuring the ADTRAN 7310-8 gateway for use with AWS to is provisioning and starting the device's packet forwarding program. To provision and start the device's packet forwarding program, follow these steps:

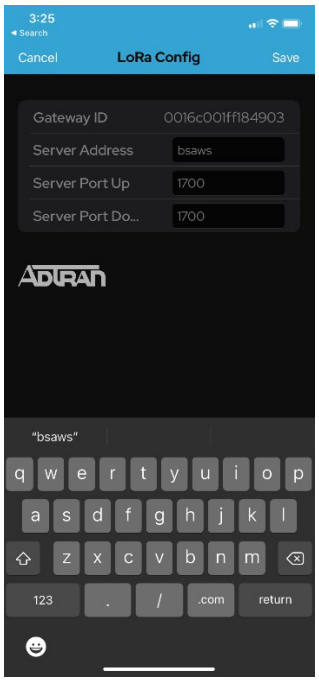
1. Open the ADTRAN IoT app on your Apple device.



2. Log into the gateway and select **LoRa Gateway Config**.



3. Enter **bsaws** into the **Server Address** field and select **Save**.



4. Your device should shortly switch to use basic station with the previously uploaded certificates.
   NOTE: *The* ***Server Port Up*** *and* ***Server Port Down*** *options are not used with Basic Station for AWS.*

7. Adding Endpoint Device(s) to the 7310-8 Gateway

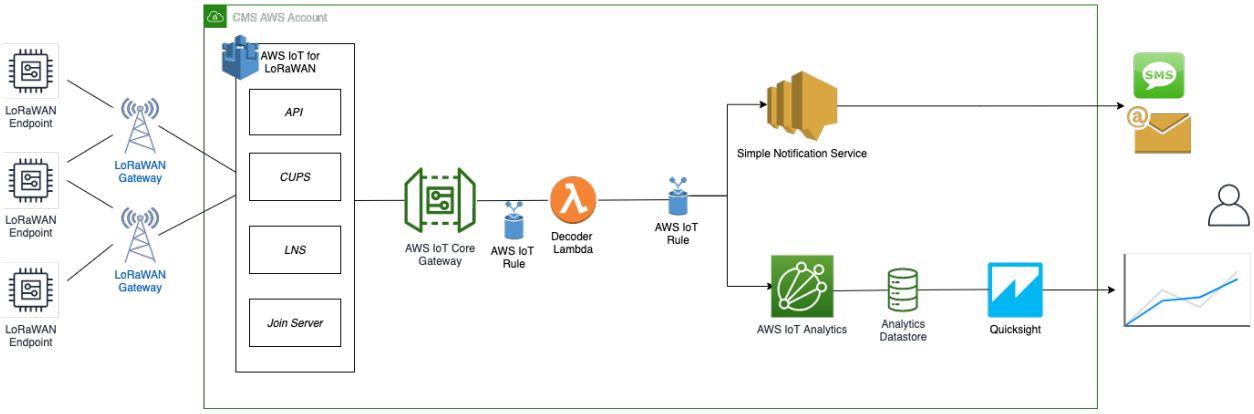
Refer to the LoRaWAN endpoint device manufacturer's documentation for configuration instructions.

8. Verifying Operation – a "Hello World" Example

Once gateway and device configuration are completed, provisioned OTAA devices can join the AWS IoT network and start to send messages. Messages from devices are received by AWS IoT Core for LoRaWAN and then forwarded to the IoT Rules Engine, which in turn facilitates communication between devices. Once a device has joined the network and can send uplink traffic, you can configure a sample "Hello World" application to test the ability of the devices to communicate across the network.

8.1 "Hello World" Sample Application Configuration Overview

The sample "Hello World" application described in this section relies on the architecture illustrated in the following network topology:



The configuration steps required to create the “Hello World” application and test a device’s ability to communicate across the IoT network includes the following:

- Creating and Testing the Lambda Function
- Creating the Destination Rule
- Configuring Amazon SNS
- Configuring IoT Analytics
- Testing your “Hello World” Application

8.2 Creating and Testing the Lambda Function

The lambda function is used to process device messages, after they have been processed by the IoT Rules Engine. The IoT Rules Engine processes these messages using a configured destination rule.

To create the lambda function to process device messages processed by the destination rule, follow these steps:

1. Go to the AWS Lambda console (console.aws.amazon.com/lambda).
2. Click on **Functions** in the navigation pane.
3. Click on **Create function**.
4. Select **Author from scratch.** Under **Basic information**, enter the function name “*sailboatdecoder*” and choose **Runtime Node.js 12.x.** from the **Runtime** drop-down menu.
5. Click on **Create function**.
6. Navigate to *<<provide your github repository URL>>* and copy the code for the lambda function.
7. Under **Function code**, paste the copied code into the editor under the **index.js** tab.
8. Once the code has been pasted, choose **Deploy** to deploy the lambda code.
9. Click on the **Permissions** tab of the lambda function
10. Change the **Lambda Role Policy** permission by following these steps:
    a. Under **Execution role**, click on the hyperlink under **Role name**.
    b. On the **Permissions** tab, find the policy name and click on it.
    c. Choose **Edit policy**, and choose the **JSON** tab.
    d. Append the following to the **Statement** section of the policy to allow publishing to AWS IoT:

```
,
{
"Effect": "Allow",
"Action": [
"iot:Publish"
],
"Resource": [
"*"
]
}
```

    e. Choose **Review Policy**, then **Save changes**.

To test the lambda function, use the following steps to create a test event:

1. In the drop-down for **Select a test event**, choose **Configure test events**.
2. Enter a name for the test event under **Event name**.
3. Paste the following sample payload in the area under **Event name**:

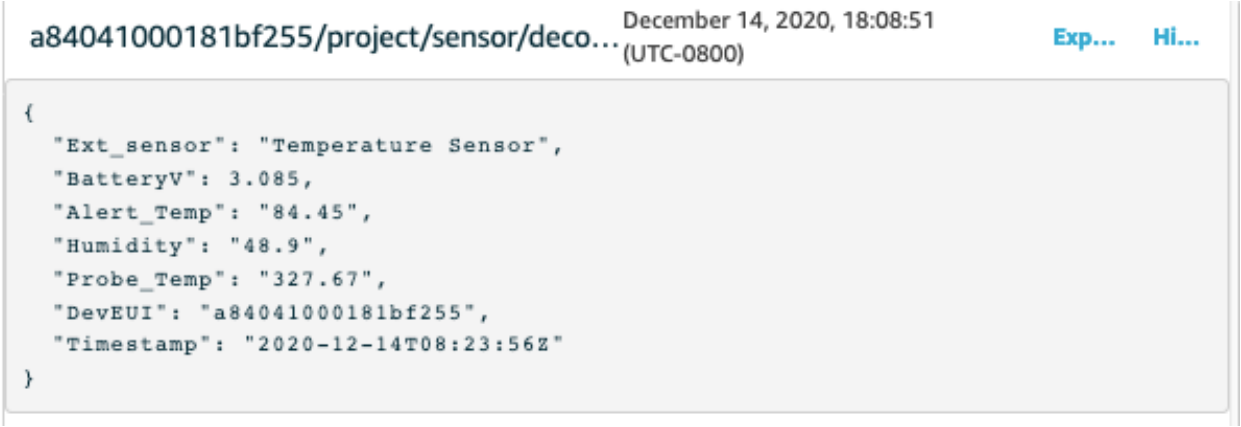
```
{
    "MessageId": "55d122ab-6355-2233-9874-ff47c5222108",
    "WirelessDeviceId": "65d128ab-90dd-4668-9556-fe47c589610b",
    "PayloadData": "zA0LYgHpAX//f/8=",
    "WirelessMetadata":
    {
        "LoRaWAN":
```

```
{
    "DevEui": "a84041000181bf255",
    "FPort": 2,
           "DataRate": 0,
           "Frequency": 904500000,
           "Gateways": [
            {
                   "GatewayEui": "80029cffXXXXXXXX",
                   "Snr": 12.25,
                   "Rssi": -47
            }
            ],
           "Timestamp": "2020-12-14T08:23:56Z",
       }
   }
}
```


4. Choose **Create** to save the event.
5. Navigate to the AWS IoT console, choose **Test** on the navigation pane, and select **MQTT client**.
6. Configure the MQTT client to subscribe to **#** (all topics).
7. Click on **Test** in the Lambda function page to generate the test event you just created.

Verify the published data in the AWS IoT Core MQTT Test client by following these steps:

1. Open another window.
2. Go to AWS IoT Console, select **Test**, under **Subscription Topic** enter **#**, and **Select** to subscribe to topic. The output should look similar to this:



8.3 Creating the Destination Rule

In this step, you create the IoT rule that forwards the device payload to your application. This rule is associated with the destination created earlier in Adding an IAM Role for the AWS IoT Core for LoRaWAN Destination.

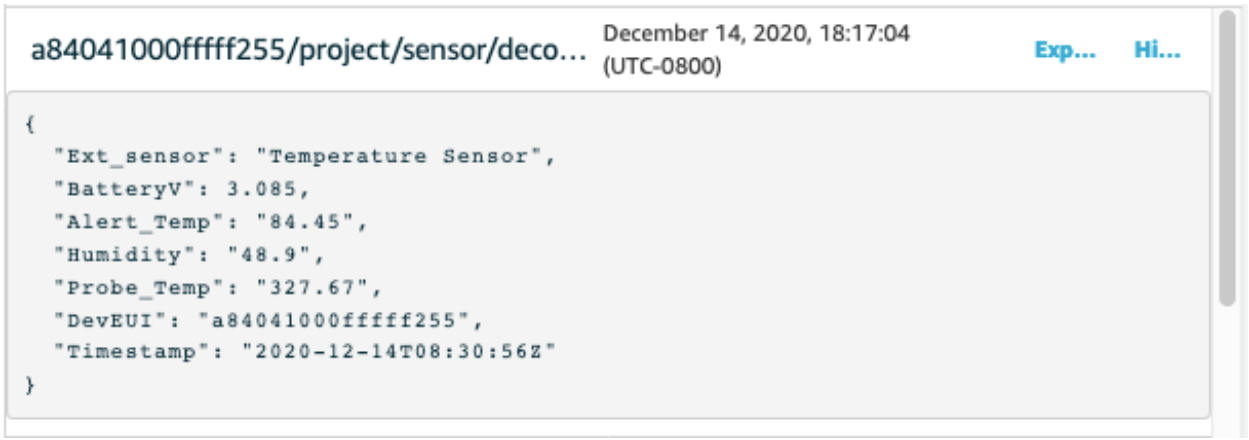
To create the destination rule, follow these steps:

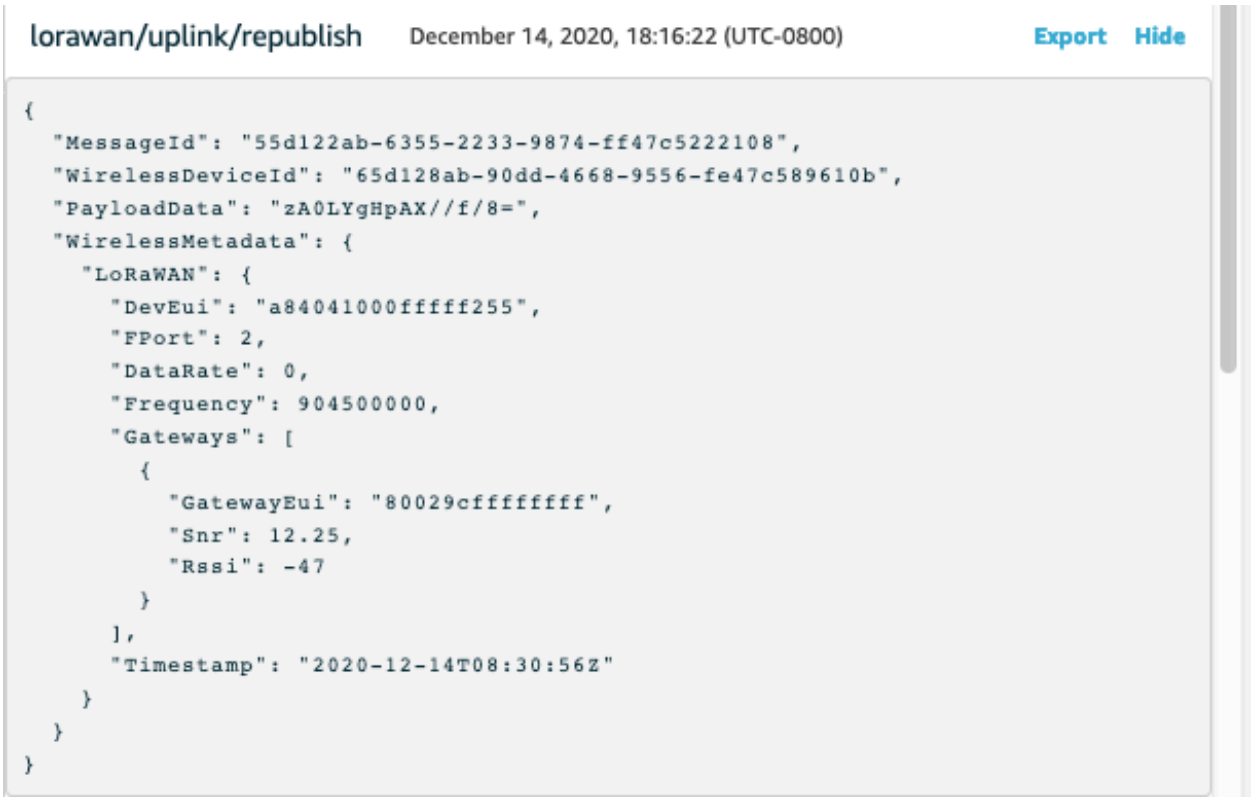
1. Navigate to the AWS IoT console.
2. In the navigation pane, choose **Act**. Then, choose **Rules**.
3. On the Rules page, choose **Create**.
4. On the **Create a rule** page, for **Name**, enter *LoRaWANRouting*. For **Description**, enter a description of your choice. Note the name of your rule. The information will be needed when you provision devices to run on AWS IoT Core for LoRaWAN.
5. Leave the default Rule query statement: '**SELECT * FROM 'iot/topic**' unchanged. This query has no effect at this time, as traffic is currently forwarded to the rules engine based on the destination.

6. Under **Set one or more actions** choose **Add action**.
7. On the **Select an action** page, choose **Republish a message to an AWS IoT topic**. Scroll down and choose **Configure action**.
8. On the Configure action page, for **Topic**, enter **project/sensor/decoded**. The AWS IoT Rules Engine will forward messages to this topic.
9. Under **Choose or create a role to grant AWS IoT access to perform this action**, choose **Create Role**.
10. For **Name**, enter a name of your choice.
11. Choose **Create role** to complete the role creation. You will see a "Policy Attached" tag next to the role name, indicating that the Rules Engine has been given permission to execute the action.
12. Choose **Add action**.
13. Add one more action to invoke the Lambda function. Under **Set one or more actions** choose **Add action**.
14. Choose **Send a message to a Lambda function**.
15. Choose **Configure action**.
16. Select the *sailboatdecoder* lambda function created earlier and choose **Add action**.
17. Then, choose **Create rule**.
18. A "Success" message will be displayed at the top of the panel, and the destination has a rule bound to it.

You can now check that the decoded data is received and republished by AWS by triggering a condition or event on the device itself. To trigger a condition or event, follow these steps:

1. Go to the AWS IoT console. In the navigation pane, select **Test**, and choose **MQTT client**.
2. Subscribe to the wildcard topic '**#**" to receive messages from all topics
3. You should see traffic similar to that shown below.





8.4 Configuring Amazon SNS

We will use the Amazon Simple Notification Service to send text messages (SMS) when certain conditions are met.

1. Go to the Amazon SNS console.
2. Click on the menu in the left corner to open the navigation pane.
3. Select **Text Messaging (SMS)** and choose **Publish text message**.
4. Under **Message type**, select **Promotional.**
5. Enter your phone number (phone number that will receive text alerts).
6. Enter "Test message" for the **Message** and choose **Publish message**.
7. If the phone number you entered is valid, you will receive a text message and your phone number will be confirmed.
8. Create an Amazon SNS Topic as follows:
    a. In the navigation pane, choose **Topics**.
    b. Select **Create topic**.
    c. Under **Details**, select **Standard**.
    d. Enter a name of your choice. Here we will use "*text_topic.*"
    e. Choose **Create topic**.
9. Create subscription for this topic as follows:
    a. In the page for the newly created *text_topic*, choose the **Subscriptions** tab.
    b. Choose **Create subscription**.
    c. Select **Protocol** as *SMS* from the drop-down.
    d. Under **Endpoint**, enter the previously validated phone number to receive the SMS alerts.
    e. Choose **Create subscription**. You should see a "*Subscription to text_topic created successfully*" message.

8.4.1 Adding a Rule for Amazon SNS Notification

To add a new rule to send an Amazon SNS notification when certain conditions are met in a decoded message, follow these steps:

1. Navigate to the AWS IoT console.
2. In the navigation pane, choose **Act**. Then, choose **Rules**.
3. On the Rules page, choose **Create**.
4. Enter the **Name** as *text_alert*, and provide an appropriate **Description**.
5. Under **Rule query statement**, enter the following query:

```
SELECT DevEUI as device_id, "Temperature exceeded 80" as message,
Alert_Temp as temp, Humidity as humidity, Timestamp as time FROM
'project/sensor/decoded' where Alert_Temp > 80
```

6. Choose **Add action**.
7. Choose **Send a message as an SNS push notification**.
8. Choose **Configure action**.
9. Under **SNS target**, select *text_topic* from the drop-down.
10. Select *RAW* under **Message format**.
11. Under **Choose or create a role to grant AWS IoT access to perform this action**, choose **Create role.**
12. Enter a name for the role and choose **Add action**.
13. Choose **Create rule**. You should see a "Success" message, indicating that the rule has been created.

8.5 Configuring IoT Analytics

We will use IoT Analytics to visually display data via graphs if there is a need in the future to do further analysis. To configure IoT analytics, you will need to create an IoT analytics rule, configure AWS IoT analytics, and configure Amazon QuickSight. The steps required to perform this configuration are provided in the following sections.

8.5.1 Creating an IoT Analytics Rule

To create an IoT analytics rule, follow these steps:

1. Navigate to the AWS IoT console.
2. In the navigation pane, choose **Act**. Then, choose **Rules**.
3. On the Rules page, choose **Create**.
4. Enter the **Name** as *Visualize*, and provide an appropriate **Description**.
5. Under **Rule query statement**, enter the following query:

```
SELECT * FROM 'project/sensor/decoded'
```

6. Choose **Add action**.
7. Select **Send a message to IoT Analytics**.
8. Choose **Configure Action**.
9. Choose **Quick Create IoT Analytics Resources**.
10. Under **Resource Prefix**, enter an appropriate prefix for your resources, such as *LoRa*.
11. Choose **Quick Create**.
12. Once the **Quick Create Finished** message is displayed, choose **Add action**.
13. Choose **Create rule**. You should see a "Success" message, indicating that the rule has been created.

8.5.2 Configuring AWS IoT Analytics

To configure AWS IoT Analytics, follow these steps:

1. Go to the AWS IoT Analytics console.
2. In the navigation panel, choose **Data sets**.
3. Select the data set that was generated by the Quick Create in Creating an IoT Analytics Rule.
4. In the **Details** section, **Edit** the **SQL query**.
5. Replace the query with:

```
select Alert_Temp as temp, Humidity as humidity, DevEUI as device_id,
Timestamp as time from LoRa_datastore
```

6. Under **Schedule**, choose **Add schedule**.

7. Under **Frequency**, choose **Every 1 minute**, and choose **Save**.

8.5.3 Configuring Amazon QuickSight

Amazon QuickSight lets you easily create and publish interactive BI dashboards that include Machine Learning-powered insights. To configure Amazon QuickSight, follow these steps:

1. Go to AWS Management console.
2. From the management console, enter "QuickSight" in the "*Search for services, features..*" search box.
3. Click on **QuickSight** in the search results.
4. If you haven't signed up for the service before, go ahead and sign up, as there is a free trial period.
5. Select the **Standard** Edition, and choose **Continue**.
6. Enter a unique name in the field **QuickSight account name**.
7. Fill in the **Notification email address**.
8. Review the other checkbox options and change them as necessary. The **AWS IoT Analytics** option must be selected.
9. Choose **Finish.** You will see a confirmation message.
10. Choose **Go to Amazon QuickSight**.
11. Select **Datasets**.
12. Select **New dataset**.
13. Select **AWS IoT Analytics**.
14. Under **Select an AWS IoT Analytics data set to import**, choose the data set created in Creating an IoT Analytics Rule.
15. Choose **Create data source**, and then choose **Visualize**.
16. Select dataset created, then select **Refresh** or **Schedule Refresh** for periodic refresh of dataset.

8.6 Testing your "Hello World" Application

Using your device, create a condition to generate an event such as a high temperature condition. If the temperature is above the configured threshold then you will receive a text alert on your phone. This alert will include key parameters about the alert.

You can also visualize the data set by following these steps:

1. Go to the AWS IoT Analytics console.
2. Choose **Data sets**.
3. Select the dataset created earlier.
4. Select **Content**. and ensure there are at least few uplink entries available in the data set.
5. Go to the QuickSight console.
6. Choose **New analysis**.
7. Choose the dataset created in Creating an IoT Analytics Rule.
8. Select time on the X-axis, Value as temp (Average) and Color as device_id to see a chart of your dataset.

9. Debugging

Debugging can be done by reviewing the logs stored on the **debug** account which can be accessed by following these steps:

1. Acquire IP address of gateway via the ADTRAN IoT app or other means.
2. **SSH** into the **debug** account of the gateway and enter the following password: **adtraniotdebug**.
3. Logs are located in **/home/debug/logs**.
4. To view AWS basic station logs: **cat home/debug/logs basic_station_aws.log**.
5. Logs are periodically saved every 30 seconds and take the last 10000 lines of program output.

10. Troubleshooting

The following are troubleshooting tips for avoiding common problems:

- You must connect the power cable to the gateway first, and then connect a serial cable to open a terminal.
- In the ADTRAN IoT app, if anything besides **bsaws** is entered into the **Server Address** field on the **LoRa Config** page, the active packet forwarding program will not be Basic Station connected to AWS.
- The **LoRa Radio Sub-Band** page on the ADTRAN IoT app is not used for AWS.

11. OTA Updates

The following outlines the OTA update/upgrade procedure:

1. Connect to the gateway via the ADTRAN IoT app.
2. Navigate to **System Status** page.
3. Select **Upgrade Device Firmware**.

   NOTE: An upgrade requires internet access and takes approximately 3 to 5 minutes to complete. Performing a firmware upgrade will move existing AWS certificates, device name, and retain the active packet forwarder; however, you will lose any statically set network settings.