

AWS MiniHub Pro

Getting Started Guide

with Basics Station & AWS IoT Core for LoRaWAN

Table of Contents

1	<i>Document Information</i>	3
1.1	About this Document	3
1.2	Naming Conventions	3
1.3	Revision History (Version, Date, Description of change)	3
2	<i>Overview</i>	3
3	<i>Hardware Description</i>	3
3.1	DataSheet	3
3.2	Standard Kit Contents	3
3.3	LED Behavior.....	4
3.4	User Provided items	4
3.5	3 rd Party purchasable items	4
3.6	Additional Hardware References	4
3.7	Reset to Default	4
3.8	Additional Software References	5
4	<i>Configure the Gateway (Web Provision)</i>	5
4.1	Connect to Web GUI	5
4.2	AWS & LoRa Setting	6
4.3	WiFi Setting	9
5	<i>Setup your AWS account and Permissions</i>	10
5.1	Overview	10
5.2	Set up Roles and Policies in IAM	10
5.2.1	Add an IAM Role for CUPS server.....	10
5.2.2	Add IAM role for Destination to AWS IoT Core for LoRaWAN.....	12
5.3	Add the Gateway to AWS IoT.....	13
5.3.1	Preparation.....	13
5.3.2	Add the LoRaWAN Gateway	13

5.4	Add a LoRaWAN Device to AWS IoT	14
5.4.1	Preparation.....	14
5.4.2	Verify Profiles.....	14
5.4.3	Set up a Destination for device traffic.....	15
5.4.4	Register the Device.....	15
6	Troubleshooting	16
7	OTA Updates	16
7.1	Get Firmware	16
7.2	Create an Amazon S3 bucket to store your update	17
7.3	Create an OTA Update service role	21
7.3.1	To create an OTA service role.....	21
7.3.2	To add OTA update permissions to your OTA service role.....	27
7.3.3	To add the required IAM permissions to your OTA service role.....	30
7.3.4	To add the required Amazon S3 permissions to your OTA service role.....	34
7.4	Create an OTA user policy	39
7.4.1	To create an OTA user policy.....	39
7.4.2	To attach the OTA user policy to your IAM user.....	48
7.4.3	Create a FreeRTOS OTA update job.....	54
8	Q&A	62
8.1	Where is the FW version info?	62
8.2	Why I cannot see the fw version info?	62
8.3	Can I disable the AWS OTA task?	62
8.4	Why I could not find the "Enable OTA" option in the "Configuration AWS & Setting" page?....	62

1 Document Information

1.1 About this Document

This document explains how to erase the MiniHub Pro flash (Model Name: TBMH110), how to upgrade new firmware, and the WiFi behavior after powering up, as well as the Web GUI usage for AWS IoT provisioning and Basic Station provisioning.

1.2 Naming Conventions

The term “downlink device” or “endpoint device” is used in this document to refer to a LoRaWAN device that connects to a LoRaWAN “Gateway”. The “Gateway” in turn, connects to AWS IoT Core for LoRaWAN.

1.3 Revision History (Version, Date, Description of change)

1.0	12-Dec-2020	Initial Version
1.1	14-Dec-2020	Second version for more information
1.2	20-Jan-2021	Remove debug board part and update setting for FW 0.9.50
1.3	28-Jan-2021	Correct some wording and hide account info based on AWS’s review request

2 Overview

The Minihub Pro is designed to enable connection to the AWS IoT Core for LoRaWAN. It is also a low-cost LoRaWAN compliant gateway utilizing a WiFi backhaul. It is a wall-plug type with interchangeable plug options. The gateway also includes a USB-C charging port that makes it ideal for mobile applications or to enlarge signal coverage.

3 Hardware Description

3.1 DataSheet

The datasheet document link <https://www.browan.com/download/kZ/stream>

3.2 Standard Kit Contents

- MiniHub Pro



3.3 LED Behavior

Colors	Blink Pattern	Mode	Status
Green	Blinking 1 sec	WIFI_STA	WiFi station not connected
Green	Blinking 1/4 sec	WIFI_STA	WiFi station connected, establishing the connection to LNS, configuring radio
Green	Solid	WIFI_STA	WiFi station connected, Sta is connected to LNS, radio listening
Green/ Orange	Blinking 1/4 sec	WIFI_STA	WiFi station connected, CUPS transaction in progress *Note: Do not unplug device in this state
Orange	Blinking 1/4 sec	CONFIG	Scanning WiFi networks, setting up configuration AP
Orange	Blinking 1 sec	CONFIG	Configuration AP active

*Note:

WIFI_STA is WiFi Station Mode
CONFIG is Configuration Mode

3.4 User Provided items

N/A

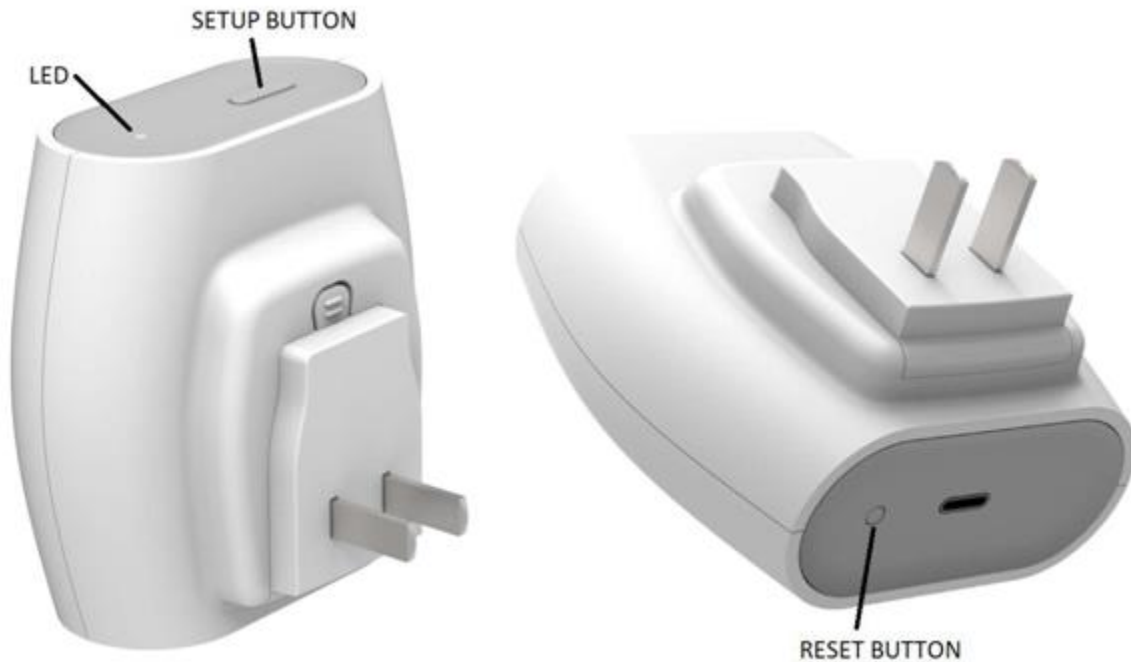
3.5 3rd Party purchasable items

N/A

3.6 Additional Hardware References

<https://www.browan.com/download/01>

3.7 Reset to Default



Press the reset button over 5 seconds to reset the system to default status. After reset to default, the orange LED will blink every 1 second.

3.8 Additional Software References

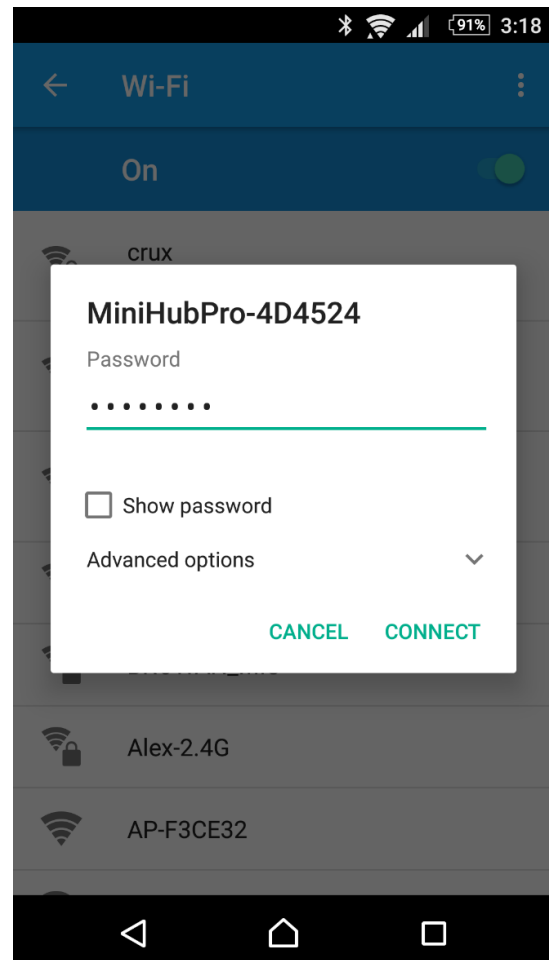
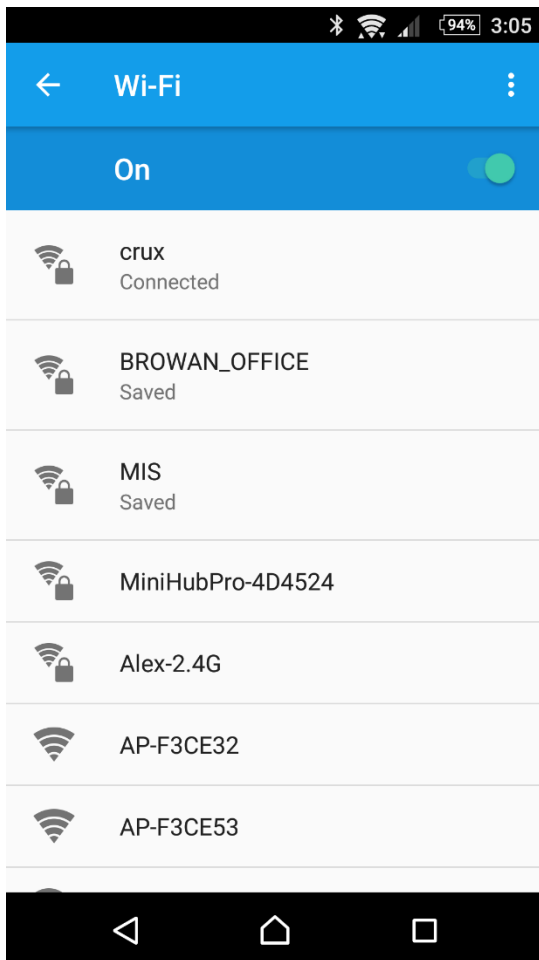
<https://www.browan.com/news/vj/detail>.

4 Configure the Gateway (Web Provision)

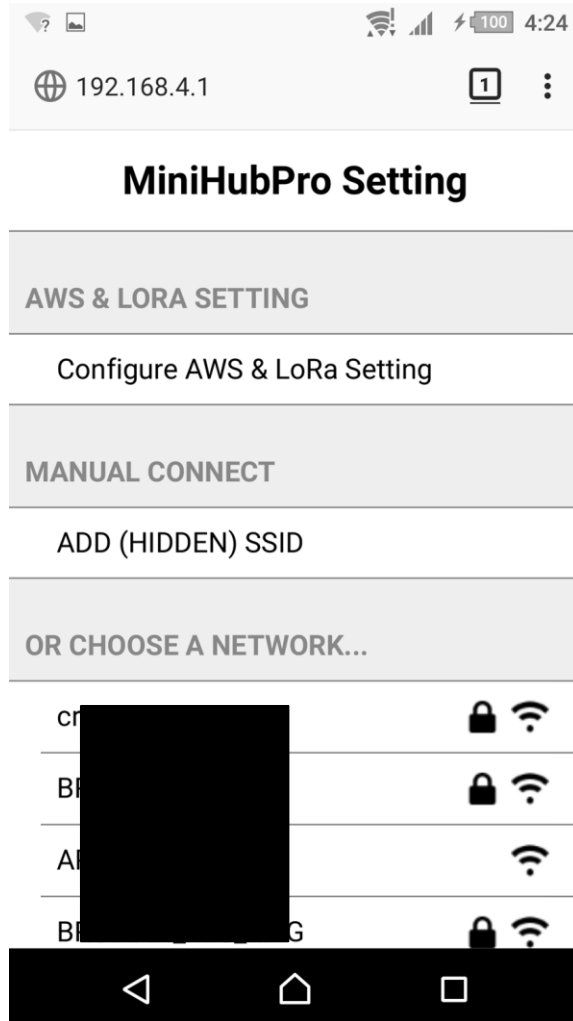
4.1 Connect to Web GUI

The device can run as WiFi AP mode or WiFi Station mode. When the device is in the initial state, such as first boot-up time or after reset-to-default. It will run in the WiFi AP mode. That means it accepts any WiFi client to connect to it.

You can find the SSID **MiniHubPro-XXXXXX** in the WiFi site-survey list. The suffix 6 characters are the last 6 hex string of WiFi MAC address. The password is **in the back label**.

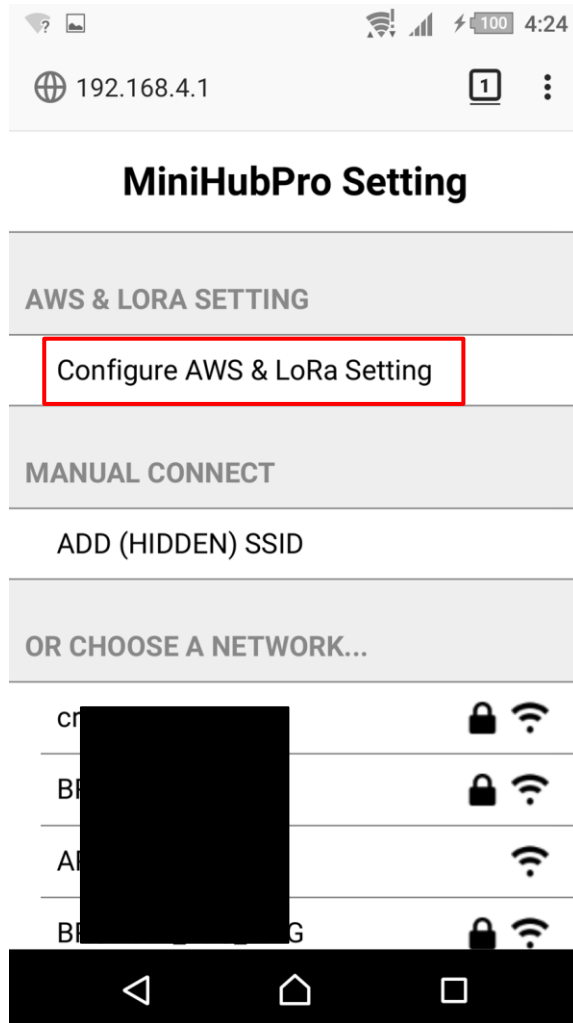


After connecting to **MiniHubPro-XXXXXX** AP, it will open the setup page. If web page doesn't open automatically, please use **Firefox or Chrome** to open **192.168.4.1** manually.



4.2 AWS & LoRa Setting

Click "Configure AWS & LoRa Setting" to open the setting page.



There are two parts, one is for AWS, and another one is for LoRa. Please configure your setting and click the "Save" button at the bottom. If you don't want to change any setting, please click the "Cancel" button at the bottom.

For AWS:

AWS Configuration is for the upgrade jobs. For more detail information, please refer to "[Section 7: OTA Updates](#)".

AWS IoT Endpoint URI: AWS IoT Custom endpoint

AWS IoT Endpoint Thing Name: The IoT thing registered on AWS

For client authentication using X509 certificates: (For FW older than 0.9.50. all credentials are encoded in DER format)

- cert is the personal certificate for the thing
- key is the personal private key for the thing

Note: For FW older than 0.9.50, please convert the pem to der file format by using below commands

```
openssl x509 -inform PEM -in Certificate.pem -outform DER -out Certificate.der  
openssl rsa -inform PEM -in PrivateKey.pem -outform DER -out PrivateKey.der
```

For LoRa:

CUPS Setting needs the information below:

CUPS URI: AWS CUPS endpoint

For client authentication using X509 certificates:

- trust is the certificate of the trusted certificate authority (CA)
- cert is the personal certificate for the gateway
- key is the personal private key for the gateway

Type:

- Boot type
Once the AP connect to the Boot CUPS, the Boot CUPS will send the arranged CUPS connection information to the AP.
- Regular type
AP will just use the Regular CUPS configuration.

Web Service: Connected.

AWS & LNS Setting

FIRMWARE VERSION

20210113_TB-300_release (0.9.50)

GATEWAY MAC

0016163002E0

AMAZON WEB SERVICES (AWS)

Enable OTA

AWS IoT Endpoint URI:

awxlj7dbw9ull-ats.iot.us-east-1.amazon

AWS IoT Endpoint Thing Name:

MiniHubPro-Test

Install Certificate (*.pem) [non-install]

Choose File 8dbfbd9a4f...ate.pem.crt

Install Private Key (*.pem) [non-install]

Choose File 8dbfbd9a4f...lic.pem.key

LORA NETWORK SERVER (LNS)

CUPS Enable:

CUPS

Type: Boot Regular

CUPS URI:

https://s2.sm.tc:7007

CUPS Trust: (installed)

Browse... No file selected.

CUPS CRT: (installed)

Browse... No file selected.

CUPS Key: (installed)

Browse... No file selected.

LNS

LNS URI:

Address:Port

LNS Trust: (non-install)

Browse... No file selected.

LNS CRT: (non-install)

Browse... No file selected.

LNS Key: (non-install)

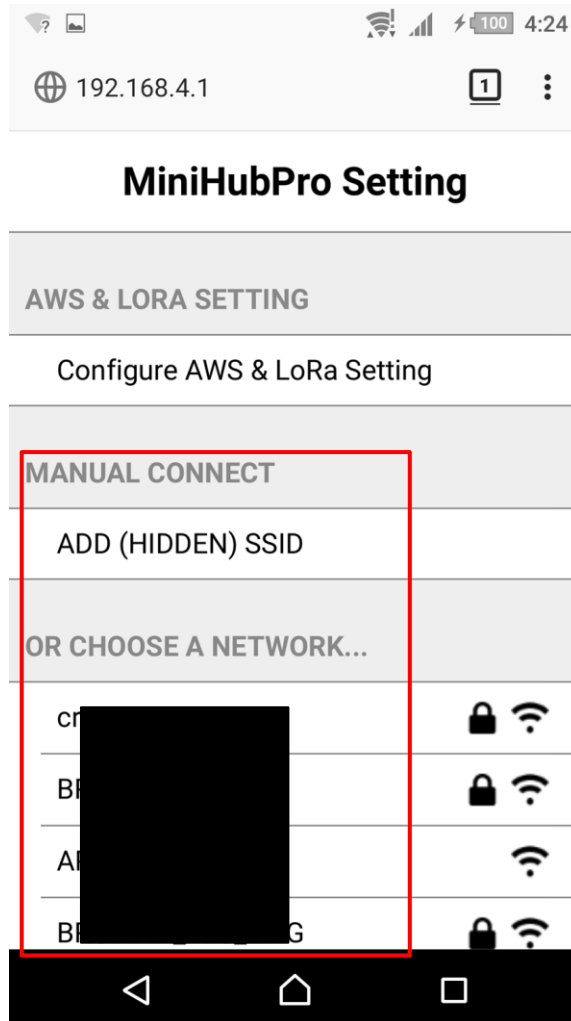
Browse... No file selected.

Cancel

Save

4.3 WiFi Setting

Choose one of the WiFi AP which you prefer to connect to the internet. You also can add SSID manually by yourself on this page. After that, the MiniHub Pro will store the connection information and switch to the WiFi Station mode.



5 Setup your AWS account and Permissions

If you don't have an AWS account, refer to the instructions in the guide [here](#). The relevant sections are **Sign up for an AWS account** and **Create a user and grant permissions**.

5.1 Overview

The high-level steps to get started with AWS IoT Core for LoRaWAN are as follows:

1. Set up Roles and Policies in IAM
2. Add a Gateway (see section [Add the Gateway to AWS IoT](#))
3. Add Device(s) (see section [Add a LoRaWAN Device to AWS IoT](#))
 - a. Verify device and service profiles
 - b. Set up a Destination to which device traffic will be routed and processed by a rule.

These steps are detailed below. For additional details, refer to the AWS [LoRaWAN developer guide](#).

5.2 Set up Roles and Policies in IAM

5.2.1 Add an IAM Role for CUPS server

Add an IAM role that will allow the Configuration and Update Server (CUPS) to handle the wireless gateway credentials.

This procedure needs to be done only once, but must be performed before a LoRaWAN gateway tries to connect with AWS IoT Core for LoRaWAN.

- Go to the [IAM Roles](#) page on the IAM console
- Choose **Create role**.
- On the **Create Role** page, choose **Another AWS account**.
- For **Account ID**, enter your account id.
- Choose **Next: Permissions**
- In the search box next to **Filter policies**, enter *AWSIoTWirelessGatewayCertManager*.
 - If the search results show the policy named *AWSIoTWirelessGatewayCertManager*, select it by clicking on the checkbox.
 - If the policy does not exist, please create it as follows:
 - Go to the [IAM console](#)
 - Choose **Policies** from the navigation pane.
 - Choose **Create Policy**. Then choose the **JSON** tab to open the policy editor. Replace the existing template with this trust policy document:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IoTWirelessGatewayCertManager",
      "Effect": "Allow",
      "Action": [
        "iot:CreateKeysAndCertificate",
        "iot:DescribeCertificate",
        "iot:ListCertificates",
        "iot:RegisterCertificate"
      ],
      "Resource": "*"
    }
  ]
}
```
 - Choose **Review Policy** to open the *Review* page.
 - For **Name**, enter *AWSIoTWirelessGatewayCertManager*. **Note** that you must not use a different name. This is for consistency with future releases.
 - For **Description**, enter a description of your choice.
 - Choose **Create policy**. You will see a confirmation message showing the policy has been created.
- Choose **Next: Tags**, and then choose **Next: Review**.
- In **Role name**, enter *IoTWirelessGatewayCertManagerRole*, and then choose **Create role**.
 - **Note** that you must not use a different name. This is for consistency with future releases.
- In the confirmation message, choose **IoTWirelessGatewayCertManagerRole** to edit the new role.
- In the **Summary**, choose the **Trust relationships** tab, and then choose **Edit trust relationship**.
- In the **Policy Document**, change the **Principal** property to represent the IoT Wireless service:

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

After you change the Principal property, the complete policy document should look like this:

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

```

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

```

- Choose **Update Trust Policy** to save your changes and exit.

At this point, you've created the *IoTWirelessGatewayCertManagerRole* and you won't need to do this again.

5.2.2 Add IAM role for Destination to AWS IoT Core for LoRaWAN

Prepare your AWS account to work with AWS IoT Core for LoRaWAN. First, create an IAM role with permissions to describe the IoT end point and to deliver messages to IoT cloud. Then, update the trust policy to grant AWS IoT Core for LoRaWAN permission to assume this IAM role when delivering messages from devices to your account.

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](#).

- In the IAM console, choose **Roles** from the navigation pane to open the **Roles** page.
- Choose **Create Role**.
- In **Select type of trusted entity**, choose **Another AWS account**.
- In **Account ID**, enter your AWS account ID, and then choose **Next: Permissions**.
- Choose **Next: Permissions**
- Search for your IAM policy. Type in the policy name to find your policy. Select it.
- Choose **Next: Tags**.
- Choose **Next: Review** to open the Review page. For **Role name**, enter an appropriate name of your choice. For **Description**, enter a description of your choice.
- Choose **Create role**.

Create the corresponding policy

- Go to the [IAM console](#)
- Choose **Policies** from the navigation pane.
- Choose **Create Policy**. Then choose the **JSON** tab to open the policy editor. Replace the existing template with this trust policy document:

```

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

```

- Choose **Review Policy** to open the Review page. For Name, enter a name of your choice. For **Description**, enter a description of your choice.
- Choose **Create policy**.

Update your policy's trust relationship.

- In the IAM console, choose **Roles** from the navigation pane to open the **Roles** page
- Enter the name of the role you created earlier in the search window, and click on the role name in the search results
- Choose the **Trust relationships** tab to navigate to the Trust relationships page.
- Choose **Edit trust relationship**. The principal AWS role in your trust policy document defaults to root. Replace the existing policy with this:

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

- Choose **Update Trust Policy**

5.3 Add the Gateway to AWS IoT

5.3.1 Preparation

To complete setting up your gateway, you need:

- 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 protocol is supported.
- Gateway ID (DevEUI) 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.
- Minimum software versions required: Basics Station 2.0.5

5.3.2 Add the LoRaWAN Gateway

To register the Gateway with AWS IoT Core for LoRaWAN, follow these steps:

- Go to the [AWS IoT Core console](#).
- Select **Wireless connectivity** in the navigation panel on the left.
- Choose **Intro**, and then choose **Get started**. This step is needed to pre-populate the default profiles.
- Under **Add LoRaWAN gateways and wireless devices**, choose **Add gateway**.
- In the **Add gateway** section, fill in the **GatewayEUI** and **Frequency band (RF Region)** fields.
- Enter a descriptive name in the **Name – optional** field. We do not recommend you leave it blank.
- Choose **Add gateway**
- On the **Configure your Gateway** page, find the section titled **Gateway certificate**.
- Select **Create certificate**.
- Once the **Certificate created and associated with your gateway** message is shown, select **Download certificates** to download the certificate (xxxxx.cert.pem) and private key (xxxxxx.private.key). ...
- In the section **Provisioning credentials**, choose **Download server trust certificates** to download the CUPS (cups.trust) and LNS (lns.trust) server trust certificates.

- Copy the CUPS and LNS endpoints and save them for use while configuring the gateway.
- Choose **Submit** to add the gateway.

5.4 Add a LoRaWAN Device to AWS IoT

5.4.1 Preparation

Locate and note the following specifications about your endpoint device.

- LoRaWAN region. This must match the gateway LoRaWAN region. The following Frequency bands (RF regions) are supported:
 - EU868
 - US915
- MAC Version. This must be one of the following:
 - V1.0.2
 - v1.0.3
 - v1.1
- OTAA v1.0x and OTAA v1.1 are supported.
- ABP v1.0x and ABP v1.1 are supported.

Locate and note the following information from your 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, NwkSkey, FNwkSIntKey, SNwkSIntKey, AppSkey

5.4.2 Verify Profiles

AWS IoT Core for LoRaWAN supports device profiles and service profiles. Device profiles contain the communication and protocol parameter values the device needs to communicate with the network server. Service profiles describe the communication parameters the device needs to communicate with the application server.

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

- Navigate to the [AWS IoT Core console](#). In the navigation pane, choose **Wireless connectivity**.
- In the navigation pane, choose **Profiles**
- In the **Device Profiles** section, there are some pre-defined profiles listed.
- Check each of the profiles to determine if one of them will work for you.
- If not, select **Add device profile** and set up the parameters as needed. For US 915 as an example, the values are:
 - MacVersion 1.0.3
 - RegParamsRevision RP002-1.0.1
 - MaxEirp 10
 - MaxDutyCycle 10
 - RfRegion US915
 - SupportsJoin true
- Continue once you have a device profile that will work for you.
- In the **Service Profiles** section, there are some pre-defined profiles listed. Check each of the profiles to determine if one of them will work for you.
- If not, select **Add service profile** and set up the parameters as needed. As an example, the default service profile parameters are shown below. However, only the AddGwMetadata setting can be changed at this time.
 - UIRate 60
 - UIBucketSize 4096

- DIRate 60
- DIBucketSize 4096
- AddGwMetadata true
- DevStatusReqFreq 24
- DrMax 15
- TargetPer 5
- MinGwDiversity 1

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

5.4.3 Set up a Destination for device traffic

Because most LoRaWAN devices don't 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 Destination. A 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.

For more information on Destinations, refer to the AWS [LoRaWAN developer guide](#).

A destination consists of a Rule and a Role. To set up the destination:

- Navigate to the [AWS IoT Core console](#). In the navigation pane, choose **Wireless connectivity**, and then **Destinations**
- Choose **Add Destination**
- On the **Add destination** page, in the **Permissions** section select the IAM role you had created earlier, from the drop-down.
- Under **Destination details** enter *ProcessLoRa* as the **Destination name**, and an appropriate description under **Destination description – optional**.

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

- For **Rule name** enter *LoRaWANRouting*. Ignore the section **Rules configuration – Optional** for now. The Rule will be set up later in the “Hello World” sample application – see [Create the IoT Rule for the destination](#)
- Choose **Add Destination**. You will see a message “*Destination added*”, indicating the destination has been successfully added.

5.4.4 Register the Device

Now register an endpoint device with AWS IoT Core for LoRaWAN as follows:

- Go to the [AWS IoT Core console](#).
- Select **Wireless connectivity** in the navigation panel on the left.
- Select **Devices**
- Choose **Add wireless device**
- On the **Add device** page, select the LoRaWAN specification version in the drop-down under **Wireless device specification**.
- Under **LoRaWAN specification and wireless device configuration**, enter the **DevEUI** and confirm it in the **Confirm DevEUI** field.
- Enter the remaining fields as per the OTAA/ABP choice you made above.
- Enter a name for your device in the **Wireless device name – optional** field.
- In the **Profiles** section, under **Wireless device profile**, find a drop-down option that corresponds to your device and region.
 - NOTE: Compare your device details to ensure the device profile is correct. If there are no valid default options, you will have to create a new profile (see the section [Verify Profiles](#)).
- Choose **Next**
- Choose the destination you created earlier (*ProcessLoRa*) from the drop-down under **Choose destination**.

- Choose **Add device**
- You will see a message saying “*Wireless device added*”, indicating that your device has been set up successfully.

6 Troubleshooting

Please see section [Q&A](#)

7 OTA Updates

7.1 Get Firmware

1. Please visit Browan's website and click release note for MiniHub Pro.
<https://www.browan.com/news/9V>
2. Download the latest MiniHub Pro firmware.
<https://www.browan.com/news/vj/detail>

Configure MiniHub Pro

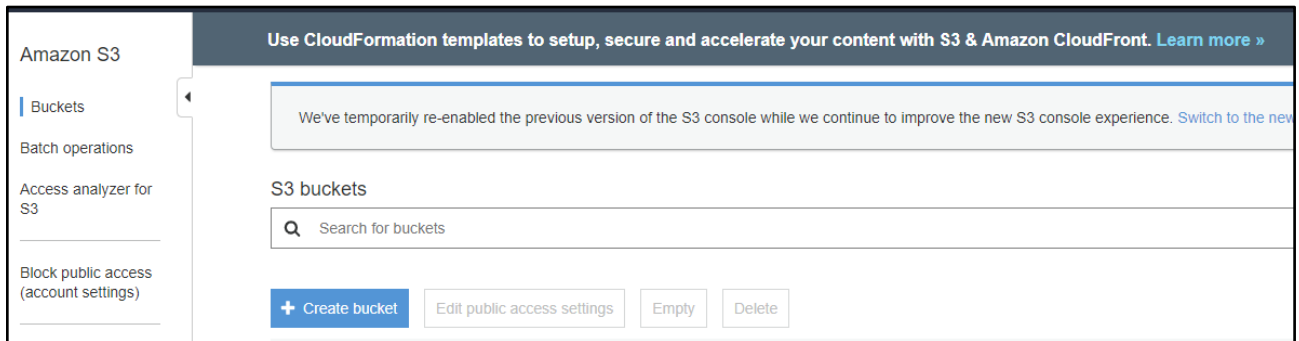
Please register things for MiniHub Pro on the AWS IoT and configure the AWS & LNS Setting.

AWS & LNS Setting

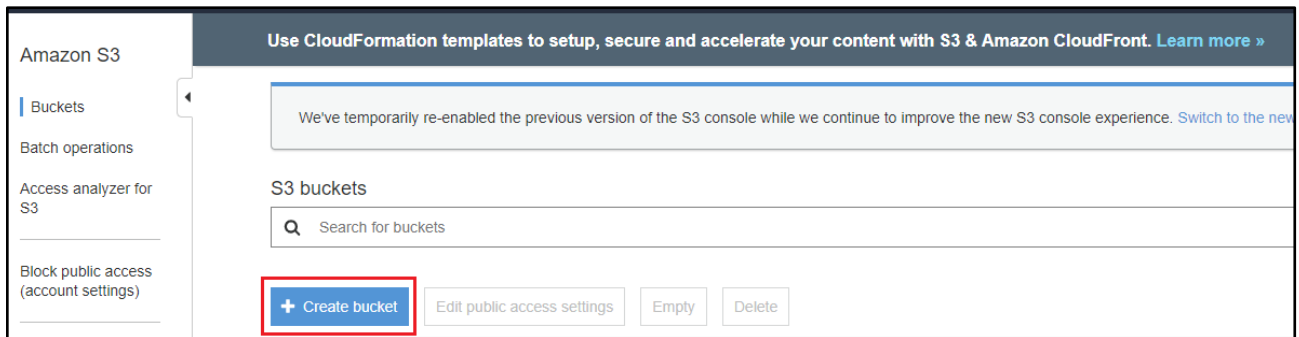
FIRMWARE VERSION
20210113_TB-300_release (0.9.50)
GATEWAY MAC
0016163002E0
AMAZON WEB SERVICES (AWS)
<input checked="" type="checkbox"/> Enable OTA
AWS IoT Endpoint URI:
<input type="text" value="https://a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6q7r8s9t0u1v2w3x4y5z6.iiot.us-east-1.amazonaws.com:8883"/>
AWS IoT Endpoint Thing Name:
<input type="text" value="MiniHubPro-Test"/>
<input checked="" type="checkbox"/> Install Certificate (*.pem) [non-install]
Choose File <input type="text" value="8dbfbd9a4f-certificate.pem.crt"/>
<input checked="" type="checkbox"/> Install Private Key (*.pem) [non-install]
Choose File <input type="text" value="8dbfbd9a4f-public.pem.key"/>

7.2 Create an Amazon S3 bucket to store your update

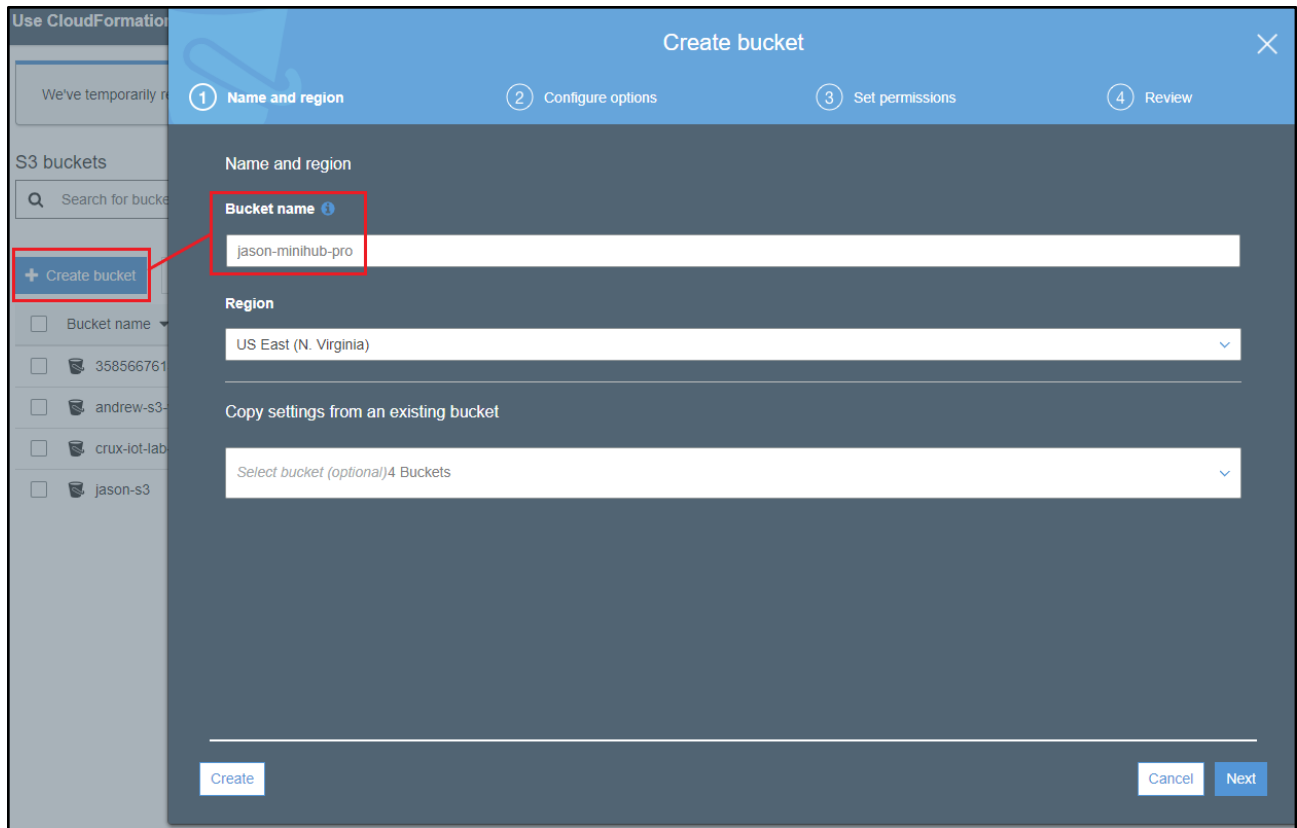
1. Sign in to the Amazon S3 console at <https://console.aws.amazon.com/s3/>



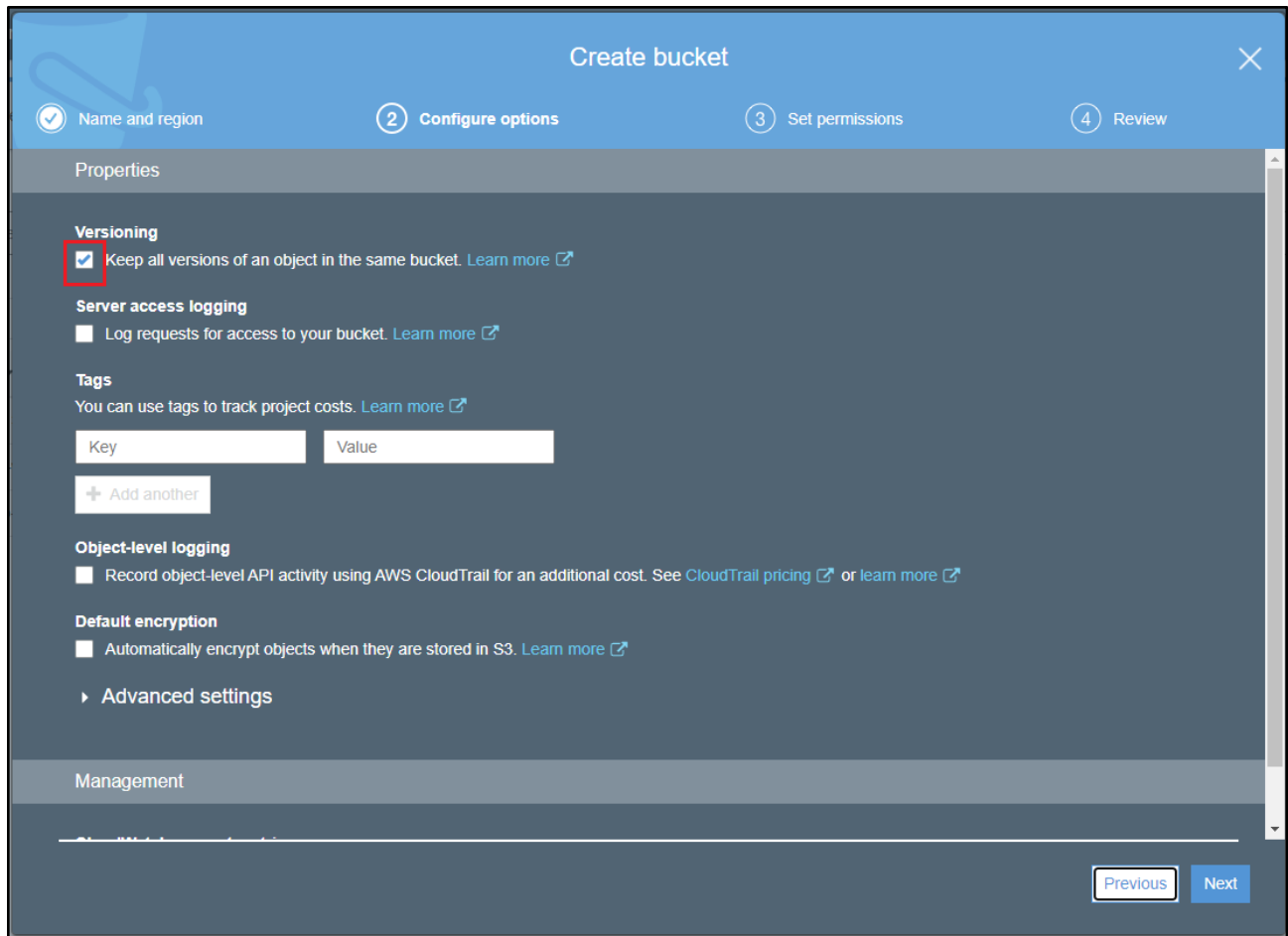
2. Choose **Create bucket**.



3. Enter a **bucket name**.



4. Under **Bucket Versioning**, select **Enable** to keep all versions in the same bucket.



5. Under **Bucket settings for Block Public Access** keep **Block all public access** selected to accept the default permissions.

Create bucket

1 Name and region 2 Configure options 3 Set permissions 4 Review

Note: You can grant access to specific users after you create the bucket.

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

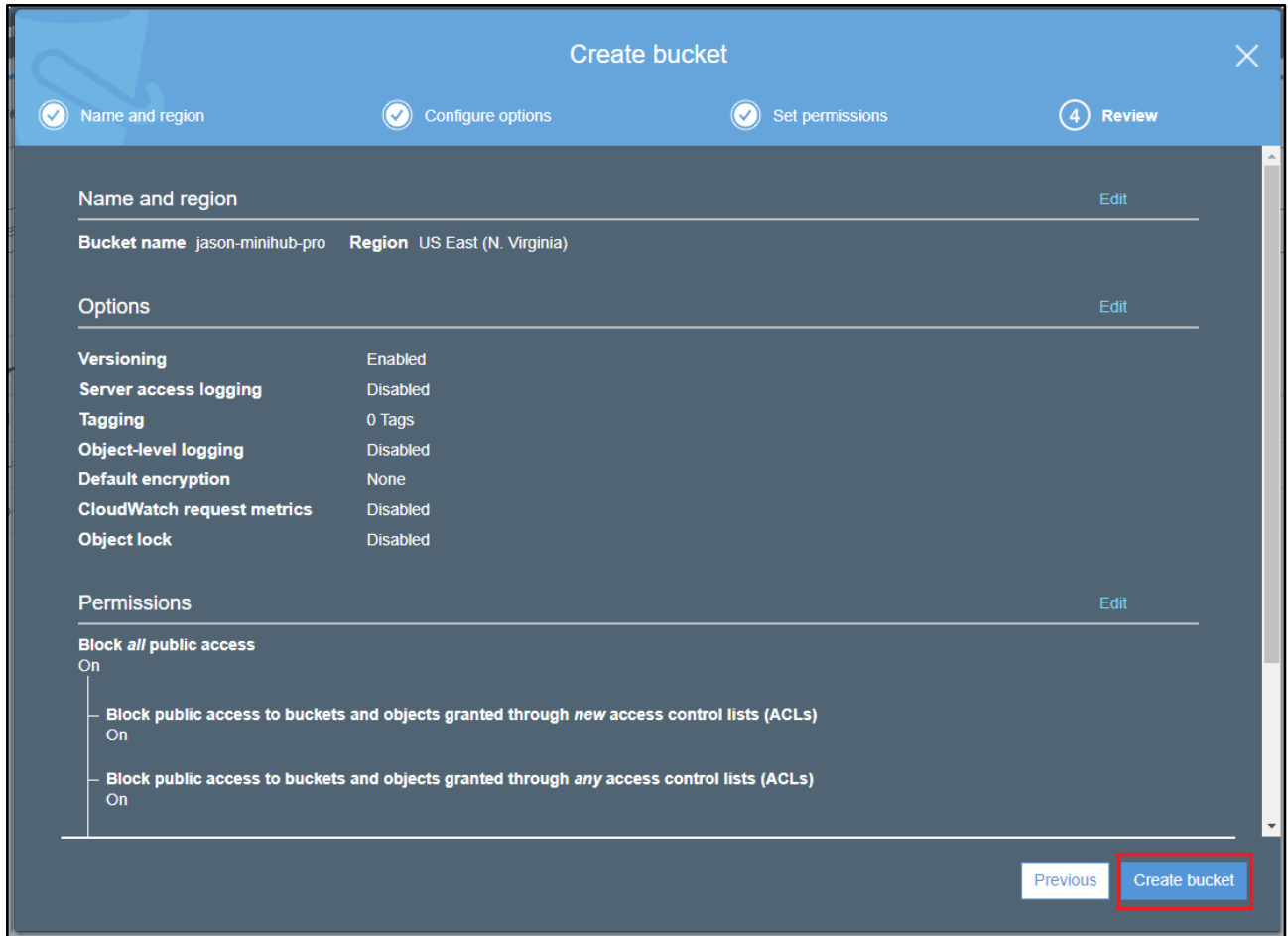
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through *new* access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through *any* access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through *new* public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Manage system permissions

Previous Next

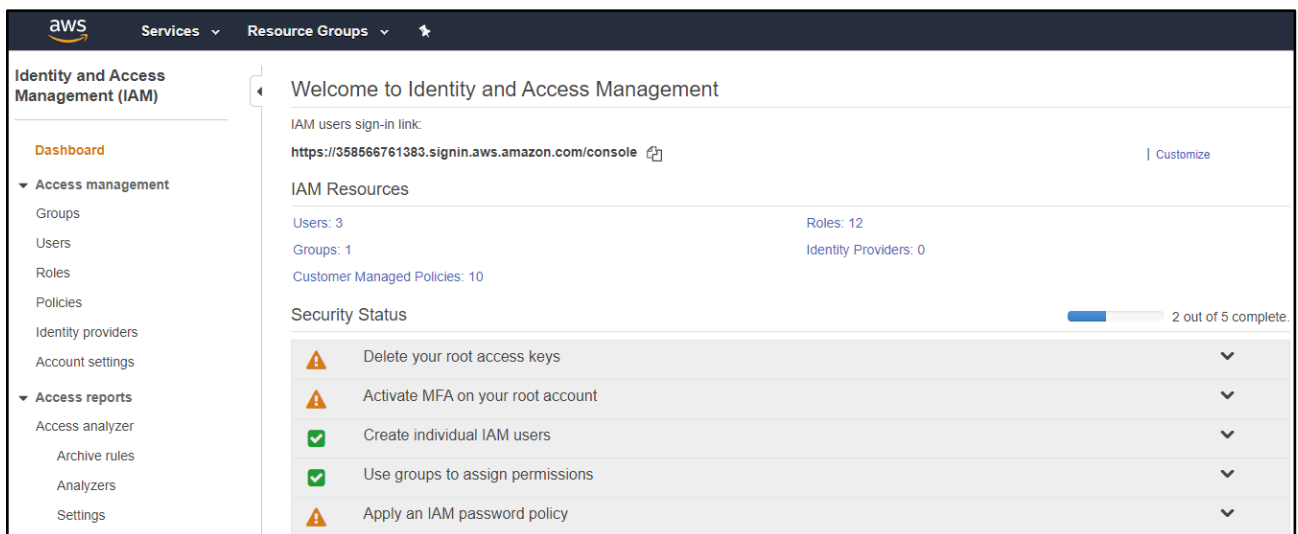
6. Choose **Create bucket**.



7.3 Create an OTA Update service role

7.3.1 To create an OTA service role

1. Sign in to the <https://console.aws.amazon.com/iam/>.



2. From the navigation pane, choose **Roles**.

The screenshot shows the AWS IAM console interface. At the top, there is a navigation bar with the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, and a home icon. The left sidebar is titled 'Identity and Access Management (IAM)' and contains a navigation menu with the following items: Dashboard, Access management (expanded), Groups, Users, Roles (highlighted in orange), Policies, Identity providers, Account settings, Access reports (expanded), Access analyzer, Archive rules, Analyzers, Settings, and Credential report. The main content area is titled 'Roles' and features a light blue informational box. This box contains the heading 'What are IAM roles?', a paragraph explaining that IAM roles are a secure way to grant permissions to trusted entities, and a bulleted list of examples: IAM user in another account, application code on an EC2 instance, an AWS service, and users from a corporate directory using SAML. Below this is a paragraph stating that IAM roles issue keys with short durations for security. Underneath is a section for 'Additional resources' with links to the IAM Roles FAQ, IAM Roles Documentation, a tutorial on setting up cross-account access, and common scenarios for roles. At the bottom of the main content area, there are two buttons: 'Create role' (blue) and 'Delete role' (grey).

3. Choose **Create role**.

Roles

What are IAM roles?

IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following:

- IAM user in another account
- Application code running on an EC2 instance that needs to perform actions on AWS resources
- An AWS service that needs to act on resources in your account to provide its features
- Users from a corporate directory who use identity federation with SAML

IAM roles issue keys that are valid for short durations, making them a more secure way to grant access.

Additional resources:

- [IAM Roles FAQ](#)
- [IAM Roles Documentation](#)
- [Tutorial: Setting Up Cross Account Access](#)
- [Common Scenarios for Roles](#)





Create role Delete role

4. Under **Select type of trusted entity**, choose **AWS Service**.

Create role

1 2 3 4

Select type of trusted entity

 AWS service EC2, Lambda and others	 Another AWS account Belonging to you or 3rd party	 Web identity Cognito or any OpenID provider	 SAML 2.0 federation Your corporate directory
--	---	---	--

Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose a use case

Common use cases

EC2
Allows EC2 instances to call AWS services on your behalf.

5. Choose **IoT** from the list of AWS services.

AWS Support	Comprehend	Elastic Container Service	Lambda	SMS
Amplify	Config	Elastic Transcoder	Lex	SNS
AppStream 2.0	Connect	ElasticLoadBalancing	License Manager	SWF
AppSync	DMS	Forecast	Machine Learning	SageMaker
Application Auto Scaling	Data Lifecycle Manager	GameLift	Macie	Security Hub
Application Discovery Service	Data Pipeline	Global Accelerator	Managed Blockchain	Service Catalog
Batch	DataSync	Glue	MediaConvert	Step Functions
Certificate Manager	DeepLens	Greengrass	Migration Hub	Storage Gateway
Chime	Directory Service	GuardDuty	OpsWorks	Systems Manager
CloudFormation	DynamoDB	Health Organizational View	Personalize	Textract
CloudHSM	EC2	IAM Access Analyzer	Purchase Orders	Transfer
CloudTrail	EC2 - Fleet	Inspector	QLDB	Trusted Advisor
CloudWatch Application Insights	EC2 Auto Scaling	IoT	RAM	VPC
CloudWatch Events	EC2 Image Builder	IoT SiteWise	RDS	WorkLink
CodeBuild	EKS	IoT Things Graph	Redshift	WorkMail

6. Under **Select your use case**, choose **IoT**.

Chime	DynamoDB	Health Organizational View	Personalize	Textract
CloudFormation	EC2	IAM Access Analyzer	Purchase Orders	Transfer
CloudHSM	EC2 - Fleet	Inspector	QLDB	Trusted Advisor
CloudTrail	EC2 Auto Scaling	IoT	RAM	VPC
CloudWatch Application Insights	EC2 Image Builder	IoT SiteWise	RDS	WorkLink
CloudWatch Events	EKS	IoT Things Graph	Redshift	WorkMail
CodeBuild				

Select your use case

IoT
Allows IoT to call AWS services on your behalf

IoT - Device Defender Audit
Provides AWS IoT Device Defender read access to IoT and related resources.

IoT - Device Defender Mitigation Actions
Provides AWS IoT Device Defender write access to IoT and related resources for execution of Mitigation Actions.

7. Choose **Next: Tags**.

Select your use case

IoT
Allows IoT to call AWS services on your behalf.

IoT - Device Defender Audit
Provides AWS IoT Device Defender read access to IoT and related resources.

IoT - Device Defender Mitigation Actions
Provides AWS IoT Device Defender write access to IoT and related resources for execution of Mitigation Actions.

* Required Cancel **Next: Permissions**

8. Choose **Next: Review**.

Add tags (optional)

IAM tags are key-value pairs you can add to your role. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this role. [Learn more](#)

Key	Value (optional)	Remove
<input type="text" value="Add new key"/>	<input type="text"/>	

You can add 50 more tags.

Cancel **Next: Review**

9. Enter a role name and description, and then choose **Create role**.

Review




Provide the required information below and review this role before you create it.

Role name* Use alphanumeric and '+=, @-_' characters. Maximum 64 characters.

Role description Maximum 1000 characters. Use alphanumeric and '+=, @-_' characters.

Trusted entities AWS service: [iot.amazonaws.com](#)

Policies

-  [AWSIoTLogging](#)
-  [AWSIoTRuleActions](#)
-  [AWSIoTThingsRegistration](#)

Permissions boundary Permissions boundary is not set

No tags were added.

* Required

[Cancel](#) [Previous](#) [Create role](#)

7.3.2 To add OTA update permissions to your OTA service role

1. In the search box on the IAM console page, enter the name of your role, and then choose it from the list.

The screenshot displays the AWS IAM console interface. On the left is a navigation sidebar for 'Identity and Access Management (IAM)' with sections for 'Access management' (Groups, Users, Roles, Policies, Identity providers, Account settings) and 'Access reports' (Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity, Service control policies (SCPs)). The main content area is titled 'Roles > jason1 Summary'. It features a metadata table with the following details:

Role ARN	arn:aws:iam:[redacted]:le/jason1
Role description	Allows IoT to call AWS services on your behalf. Edit
Instance Profile ARNs	[redacted]
Path	/
Creation time	2020-08-26 17:27 UTC+0800
Last activity	Not accessed in the tracking period
Maximum session duration	1 hour Edit

Below the metadata table are tabs for 'Permissions', 'Trust relationships', 'Tags', 'Access Advisor', and 'Revoke sessions'. The 'Permissions' tab is active, showing a section for 'Permissions policies (3 policies applied)'. An 'Attach policies' button is visible. A table lists the attached policies:

Policy name
▶ AWSIoTThingsRegistration
▶ AWSIoTLogging
▶ AWSIoTRuleActions

2. Choose **Attach policies**.

Roles > jason1

Summary

Role ARN	arn:aws:iam: [redacted] :jason1
Role description	Allows IoT to call AWS services on your behalf. Edit
Instance Profile ARNs	
Path	/
Creation time	2020-08-26 17:27 UTC+0800
Last activity	Not accessed in the tracking period
Maximum session duration	1 hour Edit

Permissions | Trust relationships | Tags | Access Advisor | Revoke sessions

▼ Permissions policies (3 policies applied)

Attach policies

Policy name ▼
▶ AWSIoTThingsRegistration
▶ AWSIoTLogging

[Show 1 more](#)

3. In the **Search** box, enter "AmazonFreeRTOSOTAUpdate", select **AmazonFreeRTOSOTAUpdate** from the list of filtered policies, and then choose **Attach policy** to attach the policy to your service role.

The screenshot shows the 'Attach Permissions' interface in the AWS IAM console. At the top, there is a 'Create policy' button. Below it, a 'Filter policies' dropdown is followed by a search box containing the text 'AmazonFreeRTOSOTAUpdate'. A table lists the filtered policies, with the first row selected. The table has columns for 'Policy name' and 'Type'. The selected row shows 'AmazonFreeRTOSOTAUpdate' with a type of 'AWS'. At the bottom right, there are two buttons: 'Cancel' and 'Attach policy', with the latter being highlighted.

Policy name	Type
AmazonFreeRTOSOTAUpdate	AWS

7.3.3 To add the required IAM permissions to your OTA service role

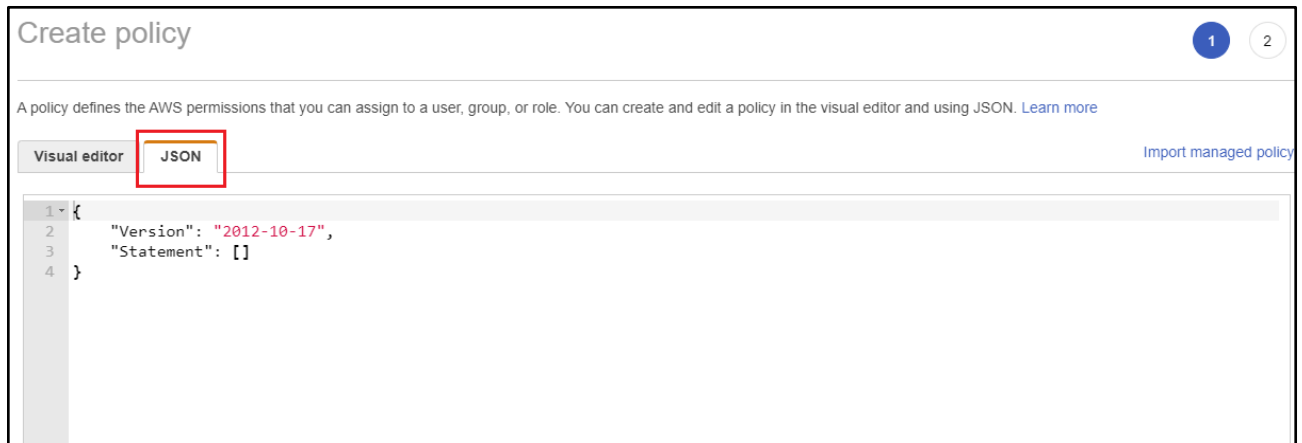
1. In the search box on the IAM console page, enter the name of your role, and then choose it from the list.

The screenshot shows the AWS IAM console interface. On the left is a navigation sidebar for 'Identity and Access Management (IAM)' with options like Dashboard, Access management, Roles, Policies, etc. The main content area is titled 'Roles > jason1 Summary'. It displays role details: Role ARN, Role description (Allows IoT to call AWS services on your behalf), Instance Profile ARNs, Path (/), Creation time (2020-08-26 17:27 UTC+0800), Last activity (Not accessed in the tracking period), and Maximum session duration (1 hour). Below this is a 'Permissions' tab with sub-tabs for Trust relationships, Tags, Access Advisor, and Revoke sessions. The 'Permissions' sub-tab is active, showing 'Permissions policies (3 policies applied)'. An 'Attach policies' button is visible, and a list of three policies is shown: AWSIoTThingsRegistration, AWSIoTLogging, and AWSIoTRuleActions.

2. Choose **Add inline policy**.

This screenshot shows the same IAM console page as above, but with an additional policy added. The 'Permissions policies' section now shows '(4 policies applied)'. The 'Attach policies' button is still present, and a red box highlights a new '+ Add inline policy' button. Below the list of policies, two rows are visible: 'AWSIoTThingsRegistration' and 'AWSIoTLogging', both identified as 'AWS managed policy' with a close icon (x) on the right. A 'Show 2 more' link is at the bottom left of the policy list.

3. Choose the **JSON** tab.



Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor **JSON** Import managed policy

```
1 {
2   "Version": "2012-10-17",
3   "Statement": []
4 }
```

4. Copy and paste the following policy document into the text box:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::your_account_id:role/your_role_name"
    }
  ]
}
```

Make sure that you replace *your_account_id* with your AWS account ID, and *your_role_name* with the name of the OTA service role.

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create an

Visual editor

JSON

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "iam:GetRole",
8         "iam:PassRole"
9       ],
10      "Resource": "arn:aws:iam:██████████:role/jason1"
11    }
12  ]
13 }
14
```


5. Choose **Review policy**.

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "iam:GetRole",
8         "iam:PassRole"
9       ],
10      "Resource": "arn:aws:iam::[REDACTED]:role/jason1"
11    }
12  ]
13 }
14
```

Character count: 148 of 10,240.
The current character count includes character for all inline policies in the role: jason1.

[Cancel](#) [Review policy](#)

6. Enter a name for the policy, and then choose **Create policy**.

Review policy

Before you create this policy, provide the required information and review this policy.

Name*

Maximum 128 characters. Use alphanumeric and +, -, @, _ characters.

Summary

Service	Access level	Resource	Request condition
Allow (1 of 238 services) Show remaining 237			
IAM	Limited: Read, Write	RoleName string like jason1	None

* Required

[Cancel](#) [Previous](#) [Create policy](#)

7.3.4 To add the required Amazon S3 permissions to your OTA service role

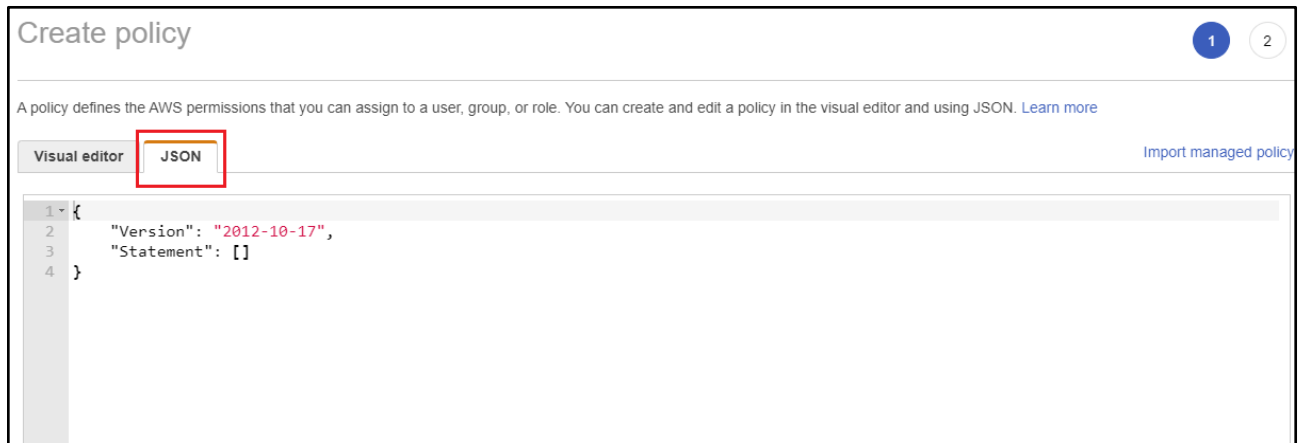
1. In the search box on the IAM console page, enter the name of your role, and then choose it from the list.

The screenshot shows the AWS IAM console interface. On the left is a navigation sidebar for 'Identity and Access Management (IAM)' with categories like 'Access management' and 'Access reports'. The main content area is titled 'Roles > jason1 Summary'. It displays key role information: Role ARN, Role description ('Allows IoT to call AWS services on your behalf.'), Instance Profile ARNs, Path, Creation time (2020-08-26 17:27 UTC+0800), Last activity, and Maximum session duration (1 hour). Below this, there are tabs for 'Permissions', 'Trust relationships', 'Tags', 'Access Advisor', and 'Revoke sessions'. The 'Permissions' tab is active, showing 'Permissions policies (3 policies applied)'. A table lists three policies: AWSIoTThingsRegistration, AWSIoTLogging, and AWSIoTRuleActions. An 'Attach policies' button is visible above the table.

2. Choose **Add inline policy**.

This screenshot shows the same IAM console page as above, but with the 'Permissions' tab expanded to show 'Permissions policies (4 policies applied)'. A table lists the existing policies: AWSIoTThingsRegistration and AWSIoTLogging, both identified as 'AWS managed policy'. A red box highlights the 'Add inline policy' button in the top right corner of the policy list area. The 'Attach policies' button is also visible.

3. Choose the **JSON** tab.



Create policy 1 2

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor **JSON** [Import managed policy](#)

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": []  
4 }
```

4. Copy and paste the following policy document into the box.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetObjectVersion",  
        "s3:GetObject",  
        "s3:PutObject"  
      ],  
      "Resource": [  
        "arn:aws:s3:::example-bucket/*"  
      ]  
    }  
  ]  
}
```

This policy grants your OTA service role permission to read Amazon S3 objects. Make sure that you replace example-bucket with the name of your bucket.



```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "s3:GetObjectVersion",
8         "s3:GetObject",
9         "s3:PutObject"
10      ],
11      "Resource": [
12        "arn:aws:s3:::jason-minihub-pro/*"
13      ]
14    }
15  ]
16 }
17
```

5. Choose Review policy.

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "s3:GetObjectVersion",
8         "s3:GetObject",
9         "s3:PutObject"
10      ],
11      "Resource": [
12        "arn:aws:s3:::jason-minihub-pro/*"
13      ]
14    }
15  ]
16 }
17
```

Character count: 316 of 10,240.
The current character count includes character for all inline policies in the role: jason1.

Cancel **Review policy**

6. Enter a name for the policy, and then choose **Create policy**.

Review policy

Before you create this policy, provide the required information and review this policy.

Name*

Maximum 128 characters. Use alphanumeric and '+-.,@-_' characters.

Summary

Service	Access level	Resource	Request condition
Allow (1 of 238 services) Show remaining 237			
S3	Limited: Read, Write	BucketName string like jason-minihub-pro, ObjectPath string like All	None

* Required

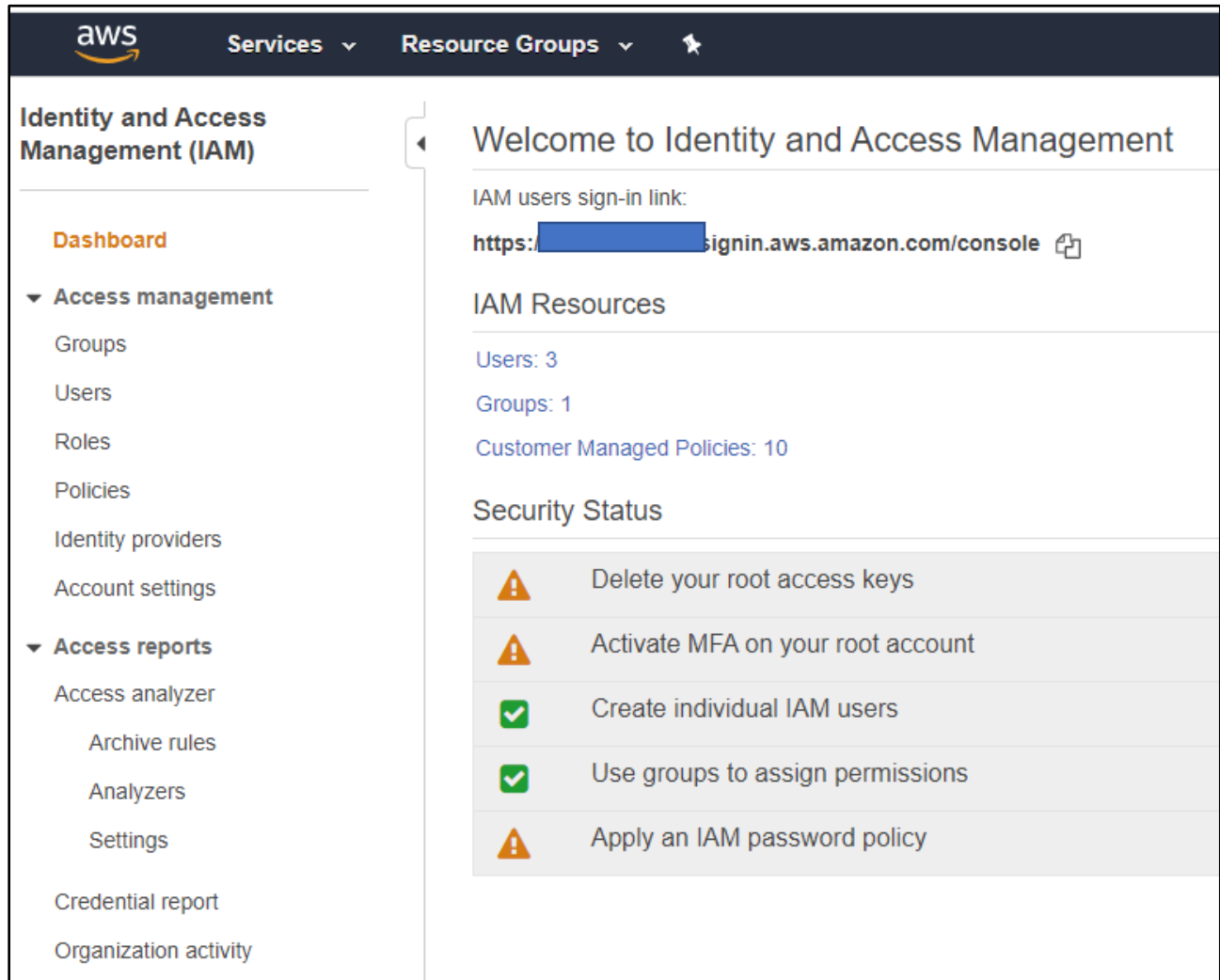
[Cancel](#) [Previous](#) [Create policy](#)

7.4 Create an OTA user policy






*If you use the "Administrator" user, you can skip this step.

7.4.1 To create an OTA user policy

1. Open the <https://console.aws.amazon.com/iam/> console.



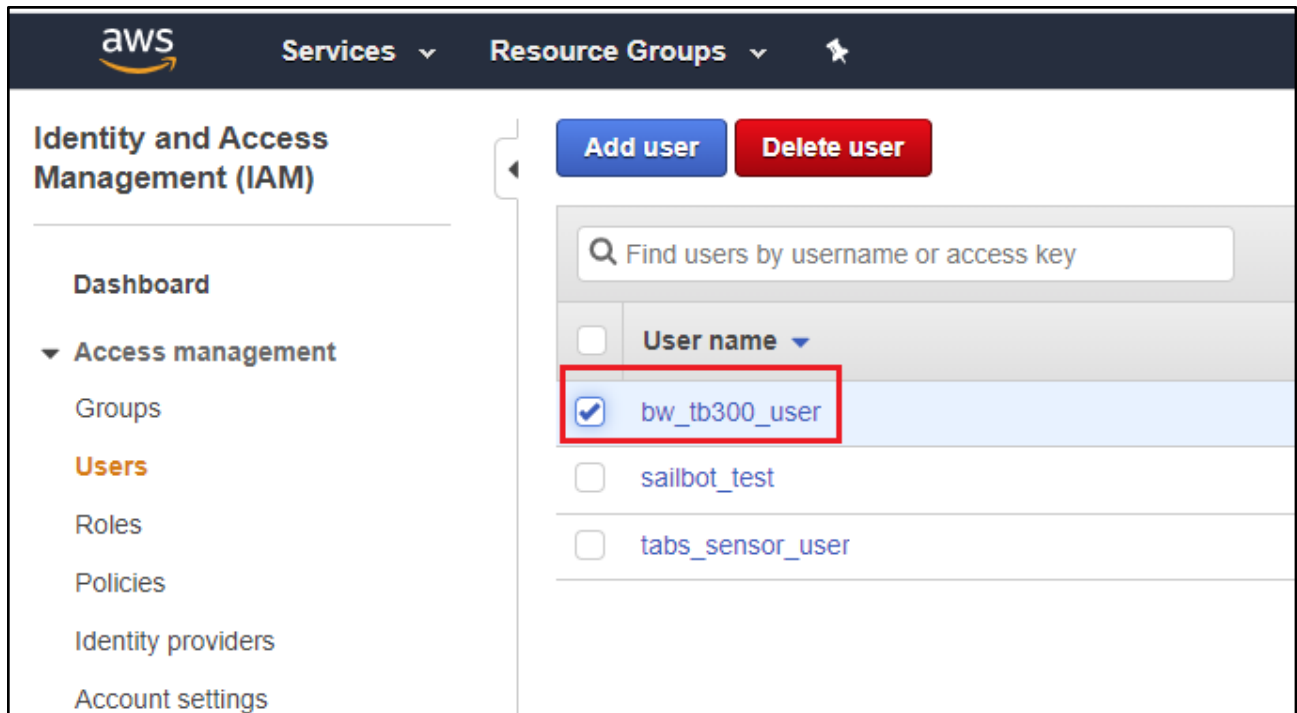
The screenshot displays the AWS IAM console interface. At the top, the AWS logo is on the left, and 'Services' and 'Resource Groups' are in the center. The left sidebar is titled 'Identity and Access Management (IAM)' and contains a 'Dashboard' link and two main sections: 'Access management' (with links for Groups, Users, Roles, Policies, Identity providers, and Account settings) and 'Access reports' (with links for Access analyzer, Archive rules, Analyzers, Settings, Credential report, and Organization activity). The main content area is titled 'Welcome to Identity and Access Management' and includes the following information:

- IAM users sign-in link:** [https://\[redacted\].signin.aws.amazon.com/console](https://[redacted].signin.aws.amazon.com/console)
- IAM Resources:**
 - Users: 3
 - Groups: 1
 - Customer Managed Policies: 10
- Security Status:**
 -  Delete your root access keys
 -  Activate MFA on your root account
 -  Create individual IAM users
 -  Use groups to assign permissions
 -  Apply an IAM password policy

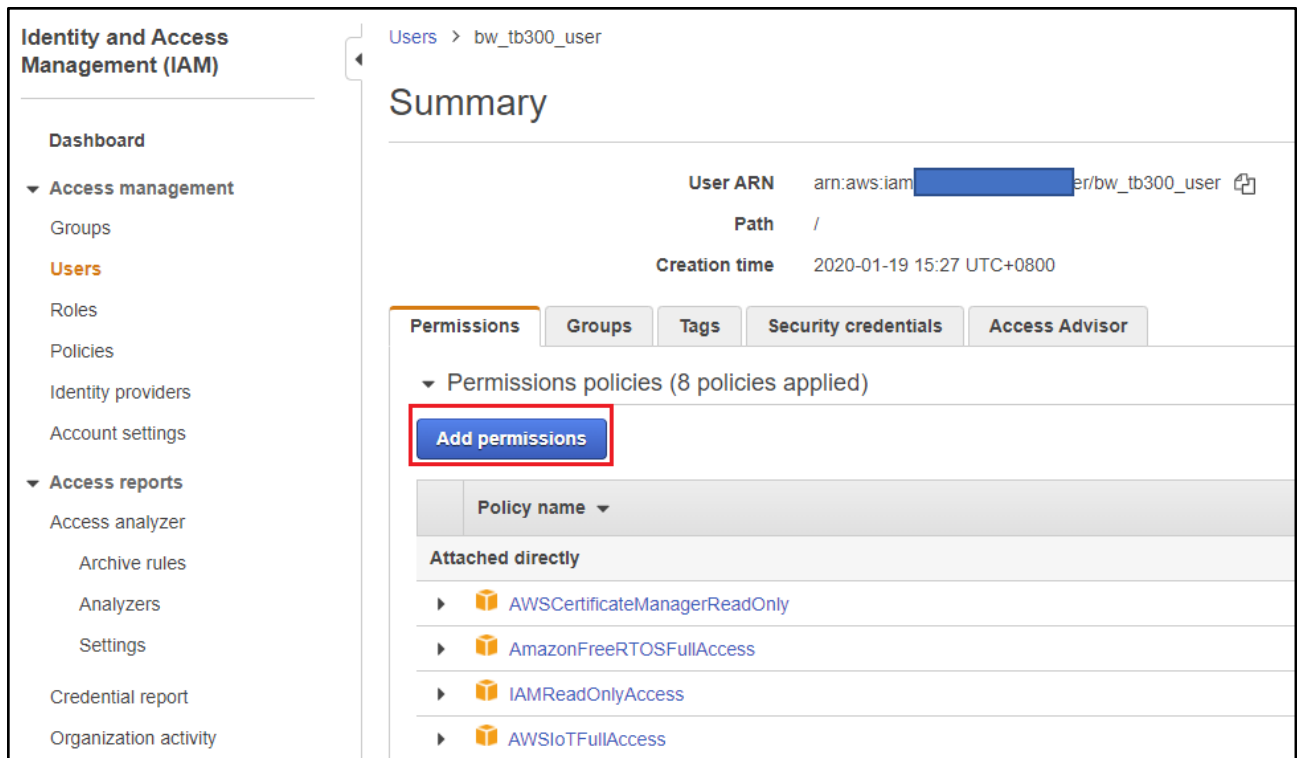
2. In the navigation pane, choose **Users**.

The screenshot displays the Identity and Access Management (IAM) console interface. On the left is a navigation pane with the following items: Dashboard, Access management (expanded), Groups, **Users** (highlighted with a red box), Roles, Policies, Identity providers, Account settings, Access reports (expanded), Access analyzer, Archive rules, and Analyzers. The main content area features a blue 'Add user' button and a red 'Delete user' button at the top. Below these is a search bar with the placeholder text 'Find users by username or access key'. A dropdown menu is open, showing 'User name' with a downward arrow. Below the dropdown, three users are listed, each with an unchecked checkbox: 'bw_tb300_user', 'sailbot_test', and 'tabs_sensor_user'.

3. Choose your IAM user from the list.



4. Choose **Add permissions**.






5. Choose **Attach existing policies directly**.

Add permissions to bw_tb300_user





Grant permissions

Use IAM policies to grant permissions. You can assign an existing policy or create a new one.

 Add user to group  Copy permissions from existing user  Attach existing policies directly

[Create policy](#)

Filter policies ▾

	Policy name ▾
<input type="checkbox"/>	 AdministratorAccess
<input type="checkbox"/>	 AlexaForBusinessDeviceSetup
<input type="checkbox"/>	 AlexaForBusinessFullAccess
<input type="checkbox"/>	 AlexaForBusinessGatewayExecution

6. Choose **Create policy**.

Add permissions to bw_tb300_user

Grant permissions

Use IAM policies to grant permissions. You can assign an existing policy or create a new one.

Add user to group Copy permissions from existing user Attach existing policies directly

Create policy

Filter policies ▾

	Policy name ▾
<input type="checkbox"/>	AdministratorAccess
<input type="checkbox"/>	AlexaForBusinessDeviceSetup
<input type="checkbox"/>	AlexaForBusinessFullAccess
<input type="checkbox"/>	AlexaForBusinessGatewayExecution

7. Choose the **JSON** tab, and copy and paste the following policy document into the policy editor:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "s3:ListAllMyBuckets",
        "s3:CreateBucket",
        "s3:PutBucketVersioning",
        "s3:GetBucketLocation",
        "s3:GetObjectVersion",
        "acm:ImportCertificate",
        "acm:ListCertificates",
        "iot:*",
        "iam:ListRoles",
        "freertos:ListHardwarePlatforms",
        "freertos:DescribeHardwarePlatform"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
```

```
        "s3:GetObject",
        "s3:PutObject"
    ],
    "Resource": "arn:aws:s3:::example-bucket/*"
},
{
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam::your-account-id:role/role-name"
}
]
}
```

Replace *example-bucket* with the name of the Amazon S3 bucket where your OTA update firmware image is stored. Replace *your-account-id* with your AWS account ID. You can find your AWS account ID in the upper right of the console. When you enter your account ID, remove any dashes (-). Replace *role-name* with the name of the IAM service role you just created.



The screenshot shows the 'Create policy' interface in the AWS IAM console. The 'JSON' tab is selected, and the following JSON code is displayed in the editor:

```
10     "s3:PutBucketVersioning",
11     "s3:GetBucketLocation",
12     "s3:GetObjectVersion",
13     "acm:ImportCertificate",
14     "acm:ListCertificates",
15     "iot:*",
16     "iam:ListRoles",
17     "freertos:ListHardwarePlatforms",
18     "freertos:DescribeHardwarePlatform"
19   ],
20   "Resource": "*"
21 },
22 {
23   "Effect": "Allow",
24   "Action": [
25     "s3:GetObject",
26     "s3:PutObject"
27   ],
28   "Resource": "arn:aws:s3:::json-minihub-pro/*"
29 },
30 {
31   "Effect": "Allow",
32   "Action": "iam:PassRole",
33   "Resource": "arn:aws:iam::[account-id]:role/jason1"
34 }
35 ]
36 }
37
```

At the bottom right of the interface, there are two buttons: 'Cancel' and 'Review policy'.

8. Choose Review policy.

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor **JSON** [Import managed policy](#)

```
10     "s3:PutBucketVersioning",
11     "s3:GetBucketLocation",
12     "s3:GetObjectVersion",
13     "acm:ImportCertificate",
14     "acm:ListCertificates",
15     "iot:*",
16     "iam:ListRoles",
17     "freertos:ListHardwarePlatforms",
18     "freertos:DescribeHardwarePlatform"
19   ],
20   "Resource": "*"
21 },
22 {
23   "Effect": "Allow",
24   "Action": [
25     "s3:GetObject",
26     "s3:PutObject"
27   ],
28   "Resource": "arn:aws:s3:::json-minihub-pro/*"
29 },
30 {
31   "Effect": "Allow",
32   "Action": "iam:PassRole",
33   "Resource": "arn:aws:iam::[REDACTED]:role/json1"
34 }
35 ]
36 }
37 }
```

[Cancel](#) [Review policy](#)

9. Enter a name for your new OTA user policy, and then choose **Create policy**.

Create policy

1 2

Review policy

Name*

Use alphanumeric and '+,=,@,_' characters. Maximum 128 characters.

Description

Maximum 1000 characters. Use alphanumeric and '+,=,@,_' characters.

Summary

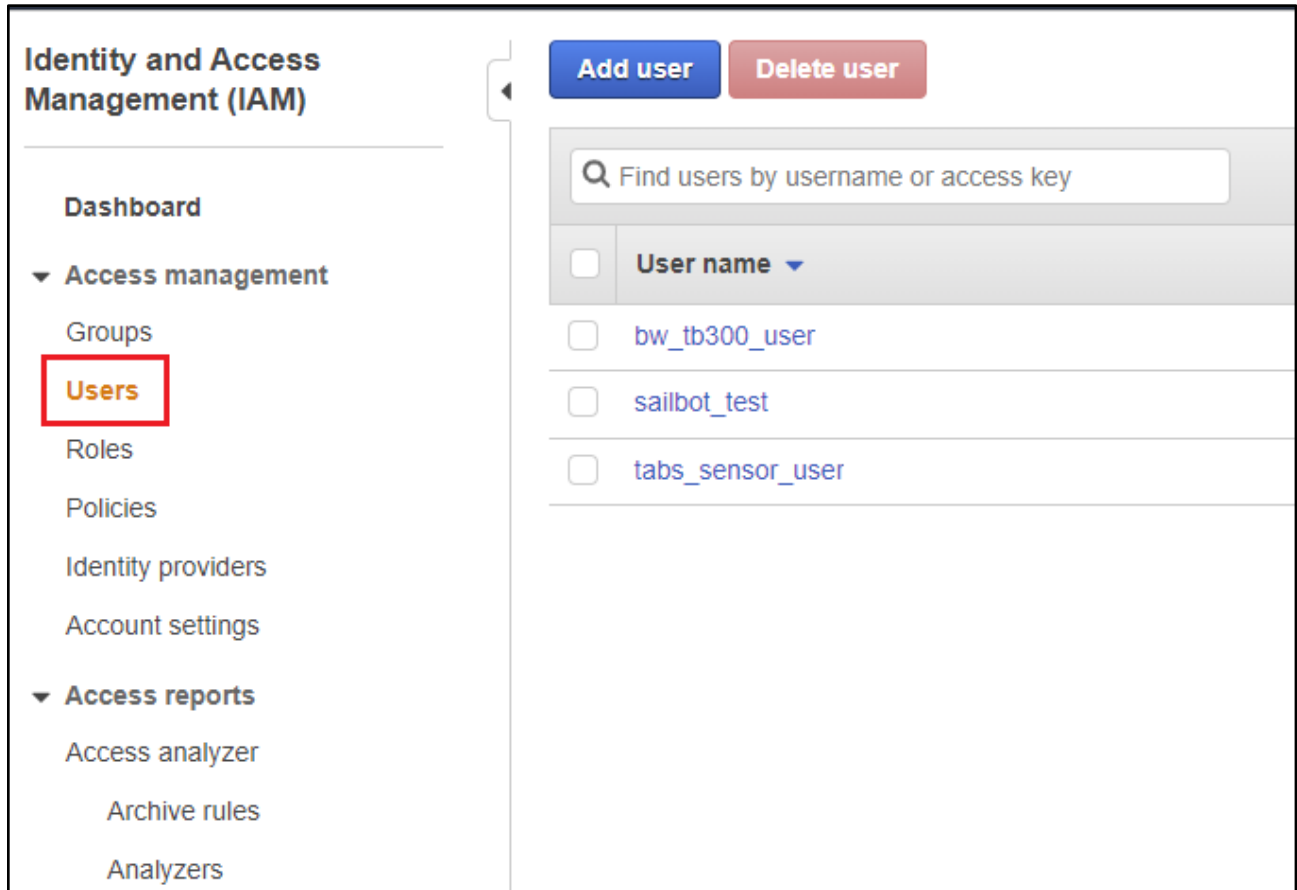
Service	Access level	Resource	Request condition
Allow (5 of 238 services) Show remaining 233			
Certificate Manager	Full: List Limited: Write	All resources	None
FreeRTOS	Limited: List, Read	All resources	None
IAM	Limited: List, Write	Multiple	None
IoT	Full access	All resources	None
S3	Limited: List, Read, Write	Multiple	None

* Required

[Cancel](#) [Previous](#) [Create policy](#)

7.4.2 To attach the OTA user policy to your IAM user

1. In the IAM console, in the navigation pane, choose **Users**, and then choose your user.



2. Choose **Add permissions**.

The screenshot displays the AWS IAM console interface for a user named 'bw_tb300_user'. The left-hand navigation pane is titled 'Identity and Access Management (IAM)' and includes sections for 'Dashboard', 'Access management' (with sub-items: Groups, Users, Roles, Policies, Identity providers, Account settings), 'Access reports' (with sub-items: Access analyzer, Archive rules, Analyzers, Settings), 'Credential report', and 'Organization activity'. The 'Users' link is highlighted in orange. The main content area shows the user's 'Summary' page with the following details:

- User ARN:** arn:aws:iam::[redacted]:user/bw_tb300_user
- Path:** /
- Creation time:** 2020-01-19 15:27 UTC+0800

Below the summary, there are five tabs: 'Permissions' (selected), 'Groups', 'Tags', 'Security credentials', and 'Access Advisor'. Under the 'Permissions' tab, a section titled 'Permissions policies (8 policies applied)' contains a blue button labeled 'Add permissions', which is highlighted with a red rectangular box. Below this button is a table with the following structure:




Policy name
Attached directly
▶ AWSCertificateManagerReadOnly
▶ AmazonFreeRTOSFullAccess
▶ IAMReadOnlyAccess
▶ AWSIoTFullAccess

3. Choose **Attach existing policies directly**.

Add permissions to bw_tb300_user





Grant permissions

Use IAM policies to grant permissions. You can assign an existing policy or create a new one.

 Add user to group  Copy permissions from existing user  Attach existing policies directly

[Create policy](#)

[Filter policies](#)

	Policy name
<input type="checkbox"/>	 AdministratorAccess
<input type="checkbox"/>	 AlexaForBusinessDeviceSetup
<input type="checkbox"/>	 AlexaForBusinessFullAccess
<input type="checkbox"/>	 AlexaForBusinessGatewayExecution

4. Search for the OTA user policy you just created and select the check box next to it.

Add permissions to bw_tb300_user

Grant permissions

Use IAM policies to grant permissions. You can assign an existing policy or create a new one.

Add user to group Copy permissions from existing user Attach existing policies directly

[Create policy](#)




Filter policies ▼

	Policy name ▼
<input type="checkbox"/>	AmazonFreeRTOSOTAUpdate
<input type="checkbox"/>	AWSIoTAnalyticsFullAccess
<input type="checkbox"/>	AWSIoTAnalyticsReadOnlyAccess
<input type="checkbox"/>	AWSIoTOTAUpdate
<input type="checkbox"/>	AWSQuickSightIoTAnalyticsAccess
<input type="checkbox"/>	GreengrassOTAUpdateArtifactAccess
<input type="checkbox"/>	IoTAccess_Sailboat
<input checked="" type="checkbox"/>	jason_OTA

5. Choose **Next: Review**.







Grant permissions

Use IAM policies to grant permissions. You can assign an existing policy or create a new one.

 Add user to group  Copy permissions from existing user  Attach existing policies directly

Create policy

Filter policies

	Policy name	Type
<input type="checkbox"/>	 AmazonFreeRTOSOTAUpdate	AWS
<input type="checkbox"/>	 AWSIoTAnalyticsFullAccess	AWS
<input type="checkbox"/>	 AWSIoTAnalyticsReadOnlyAccess	AWS
<input type="checkbox"/>	 AWSIoTOTAUpdate	AWS
<input type="checkbox"/>	 AWSQuickSightIoTAnalyticsAccess	AWS
<input type="checkbox"/>	 GreengrassOTAUpdateArtifactAccess	AWS
<input type="checkbox"/>	IoTAccess_Sailboat	Cust
<input checked="" type="checkbox"/>	jason_OTA	Cust

[Cancel](#) [Next: Review](#)

6. Choose **Add permissions**.

Add permissions to bw_tb300_user

Permissions summary

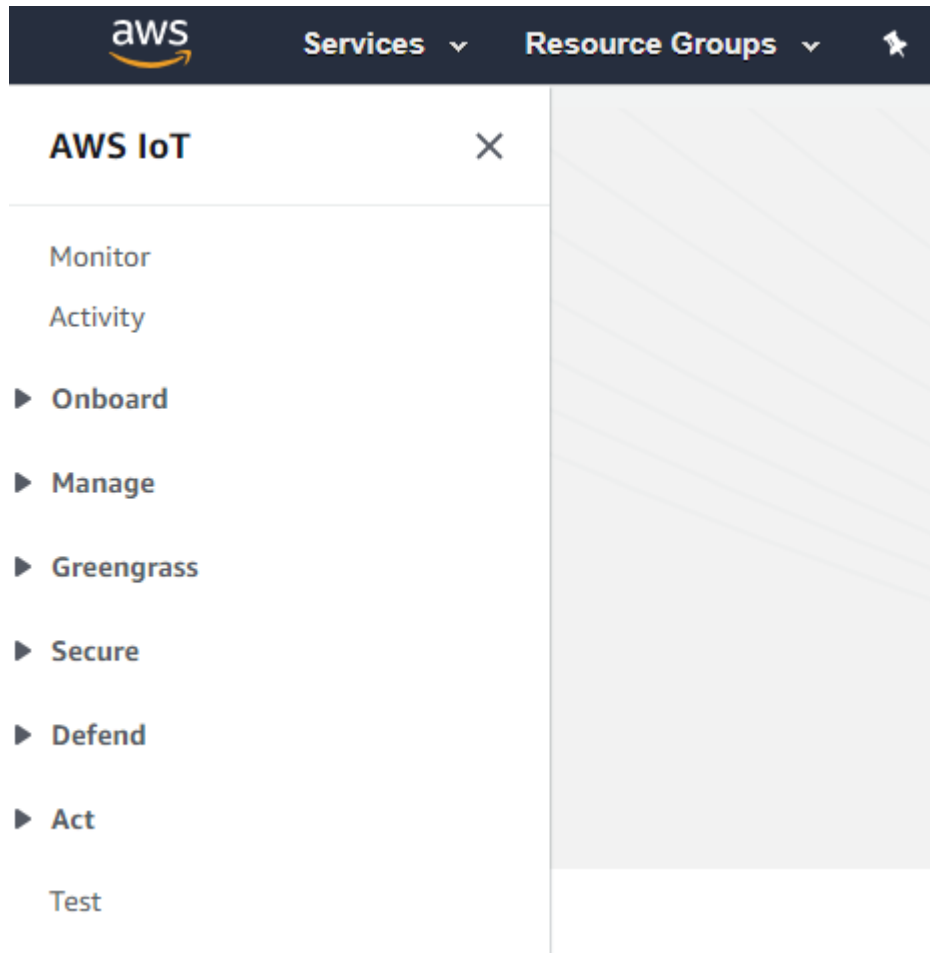
The following policies will be attached to the user shown above.

Type	Name
Managed policy	jason_OTA

Cancel Previous **Add permissions**

7.4.3 Create a FreeRTOS OTA update job

1. Go to "IoT Core" service.



2. Go to "Manage Jobs" and click the "Create" button.

The screenshot shows the AWS IoT Jobs page. The top navigation bar includes the AWS logo, "Services" with a dropdown arrow, and "Resource Groups" with a dropdown arrow and a star icon. The left sidebar is titled "AWS IoT" and contains a list of navigation options: Monitor, Activity, Onboard, Manage, Things, Types, Thing groups, Billing groups, Jobs (highlighted in orange), Tunnels, and Greengrass. The main content area is titled "Jobs" and features a search bar labeled "Search jobs" with a magnifying glass icon. Below the search bar, there are two job cards. The first card is titled "AFR_OTA-otaTest_001" and shows "SNAPSHOT COMPLETED" with two filled circles. The second card is titled "AFR_OTA-cruX_v" and also shows "SNAPSHOT COMPLETED" with two filled circles.

3. Select "Create OTA update job"

[AWS IoT](#) > [Jobs](#) > Create job

CREATE JOB

Select a job

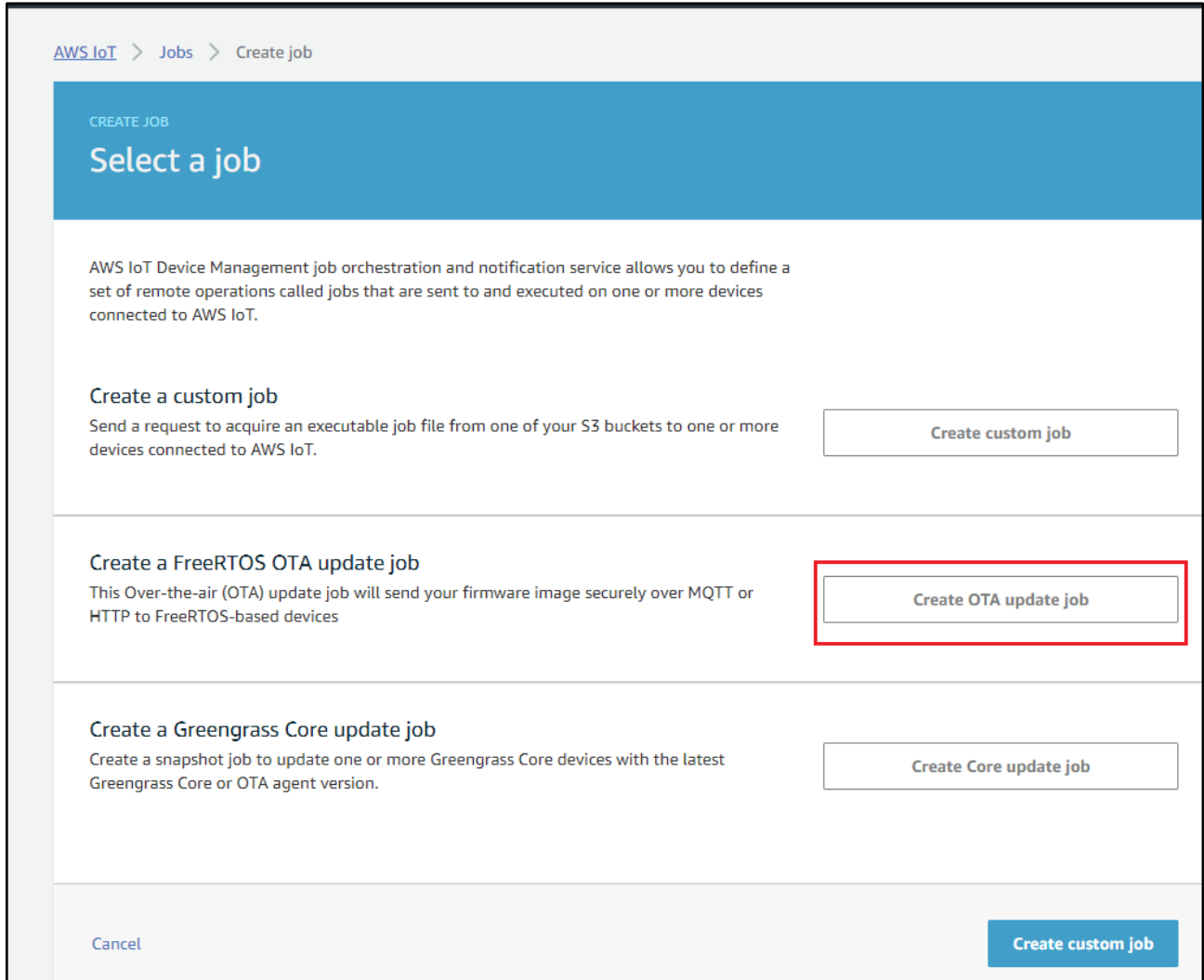
AWS IoT Device Management job orchestration and notification service allows you to define a set of remote operations called jobs that are sent to and executed on one or more devices connected to AWS IoT.

Create a custom job
Send a request to acquire an executable job file from one of your S3 buckets to one or more devices connected to AWS IoT.

Create a FreeRTOS OTA update job
This Over-the-air (OTA) update job will send your firmware image securely over MQTT or HTTP to FreeRTOS-based devices

Create a Greengrass Core update job
Create a snapshot job to update one or more Greengrass Core devices with the latest Greengrass Core or OTA agent version.

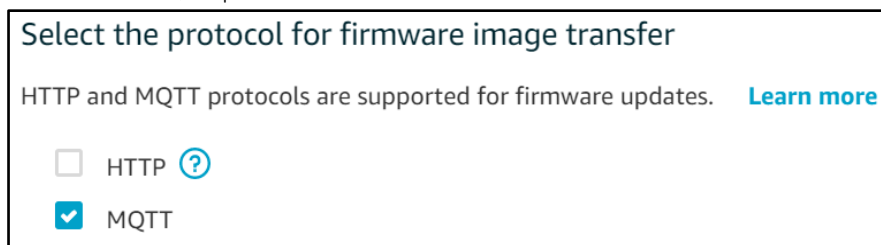
Cancel Create custom job



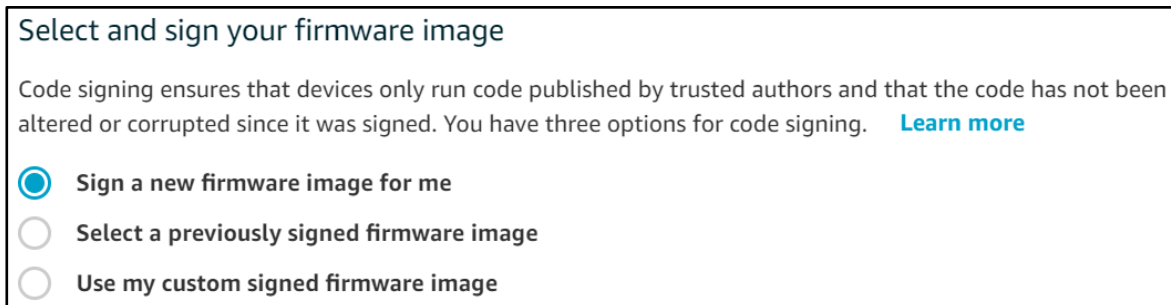
4. Select the things name which configured to MiniHub Pro. And click "Next".



5. Select the "MQTT" protocol



6. Select the "Sign a new firmware image for me."



7. Create a new Code signing profile



- Click "Create"
- Input the "Profile name"
- Select hardware platform: **ESP-WROVER-KIT**
- Import the "Certificate"

Certificate:

<https://drive.google.com/file/d/1SFUXI1uqm3OWOhDs5TyDo62jqlksmGO/view?usp=sharing>

Certificate private key:

<https://drive.google.com/file/d/1EavG36gmL3cdkQxqTrTZIWTjDPm4Mmz4/view?usp=sharing>

- Input the Pathname of code signing certificate on device: P11_CSK
- Click "Create"

*Next time you can select this profile directly.

8. Upload the firmware

Select your firmware image in S3 or upload it

Image not selected
Select

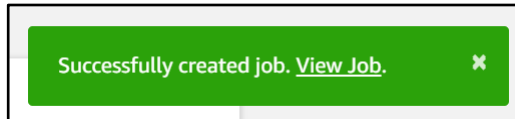
- Click "Select"
- Choose the bucket which store the firmware image.
- Click "Upload an image" Upload the image file: aws_demos.bin

Input the pathname of firmware image on the device: P11_CSK

Select "IAM role for OTA update job" (Created at step.2)

Input the OTA job unique ID and click the "Create" button.

You can find the successfully created job message.



The OTA job status is "Queued"

JOB

AFR_OTA-miniHubPro_ota_demo_0001

Actions ▾

Overview

Last updated Jun 16, 2020 8:24:41 PM +0800

All Statuses Refresh

	1 Queued	0 In progress	0 Timed out	0 Failed	0 Succeeded	0 Rejected	0 Canceled	0 Removed
--	-------------	------------------	----------------	-------------	----------------	---------------	---------------	--------------

Resource	Last updated	Status
> MiniHubPro-3D807C	Jun 16, 2020 8:24:38 PM +0800	Queued ***

Now please power on the MiniHub Pro. To trigger the OTA job process.

Current App Version: 20200601_TB-300_release

```
* Application information:
* Project name:      esp-idf
* App version:      20200601_TB-300_release
* Compile time:     Jun  1 2020 17:55:15
* ELF file SHA256:  6208adb82674992c...
* ESP-IDF:         v3.3-163-g601a03e
```

Current OTA Version: 0.9.2

```
12 621 [iot_thread] INFO: NVS> iot_thing_name = [MiniHubPro-3D807C]
13 621 [iot_thread] OTA Version 0.9.2
```

Start OTA Job:

```
28 1111 [OTA Agent Task] [prvOTAAgentTask] Called handler. Current State [Ready] Event
[Start] New state [RequestingJob]
29 1111 [OTA Agent Task] [INFO ][MQTT][11110] (MQTT connection 0x3ffef418) SUBSCRIBE
operation scheduled.
30 1111 [OTA Agent Task] [INFO ][MQTT][11110] (MQTT connection 0x3ffef418, SUBSCRIBE
operation 0x3ffffbbbc) Waiting for operation completion.
31 1121 [OTA Agent Task] [INFO ][MQTT][11210] (MQTT connection 0x3ffef418, SUBSCRIBE
operation 0x3ffffbbbc) Wait complete with result SUCCESS.
32 1121 [OTA Agent Task] [prvSubscribeToJobNotificationTopics] OK: $aws/things/MiniHubPro-
3D807C/jobs/$next/get/accepted
33 1121 [OTA Agent Task] [INFO ][MQTT][11210] (MQTT connection 0x3ffef418) SUBSCRIBE
operation scheduled.
34 1121 [OTA Agent Task] [INFO ][MQTT][11210] (MQTT connection 0x3ffef418, SUBSCRIBE
operation 0x3ffffbbbc) Waiting for operation completion.
35 1131 [OTA Agent Task] [INFO ][MQTT][11300] (MQTT connection 0x3ffef418, SUBSCRIBE
operation 0x3ffffbbbc) Wait complete with result SUCCESS.
36 1131 [OTA Agent Task] [prvSubscribeToJobNotificationTopics] OK: $aws/things/MiniHubPro-
3D807C/jobs/notify-next
37 1131 [OTA Agent Task] [prvRequestJob_Mqtt] Request #0
38 1131 [OTA Agent Task] [INFO ][MQTT][11310] (MQTT connection 0x3ffef418) MQTT PUBLISH
operation queued.
39 1131 [OTA Agent Task] [INFO ][MQTT][11310] (MQTT connection 0x3ffef418, PUBLISH
operation 0x3ffffbbbc) Waiting for operation completion.
40 1138 [OTA Agent Task] [INFO ][MQTT][11380] (MQTT connection 0x3ffef418, PUBLISH
operation 0x3ffffbbbc) Wait complete with result SUCCESS.
41 1138 [OTA Agent Task] [prvOTAAgentTask] Called handler. Current State [RequestingJob]
Event [RequestJobDocument] New state [WaitingForJob]
42 1139 [OTA Agent Task] [prvParseJobDoc] Size of OTA_FileContext_t [64]
43 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ clientToken:
0:MiniHubPro-3D807C ]
44 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ jobId: AFR_OTA-
minihubpro_ota_demo_0001 ]
45 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ protocols: ["MQTT"] ]
46 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ streamname: AFR_OTA-
7bd6fc8c-d14f-4a3c-8789-08aee20d6cc4 ]
47 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ filepath: P11_CSK ]
```

```
48 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ filesize: 1312384 ]
49 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ fileid: 0 ]
50 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ certfile: P11_CSK ]
51 1139 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ sig-sha256-ecdsa:
MEUCIQC11uQ8fw/5qJbMeVJYGVbvXULR... ]
52 1139 [OTA Agent Task] [prvParseJobDoc] Job was accepted. Attempting to start transfer.
```

Downloaded and verified:

```
1182 3338 [OTA Agent Task] [prvIngestDataBlock] Received final expected block of file.
1183 3338 [OTA Agent Task] [prvStopRequestTimer] Stopping request timer.
1184 3341 [OTA Agent Task] [INFO ][DEMO][33410] Entering get_item_from_nvs with [P11_CSK]
1185 3341 [OTA Agent Task] [INFO ][DEMO][33410] Non-Volatile Storage (NVS) handle...[22]
1186 3341 [OTA Agent Task] [INFO ][DEMO][33410] Length of the [P11_CSK] is: [365]
1187 3351 [OTA Agent Task] [INFO ][DEMO][33510] Leaving get_item_from_nvs
1188 3351 [OTA Agent Task] [prvIngestDataBlock] File receive complete and signature is
valid.
1189 3351 [OTA Agent Task] [prvStopRequestTimer] Stopping request timer.
1190 3351 [OTA Agent Task] [prvUpdateJobStatus_Mqtt] Msg:
{"status":"IN_PROGRESS","statusDetails":{"self_test":"ready","updatedBy":"0x90002"}}
```

Upgraded App Version: 20200616_TB-300_release

```
* Application information:
* Project name:      esp-idf
* App version:      20200616_TB-300_release
* Compile time:     Jun 16 2020 11:31:12
* AWS APP Version:  0.9.4
* ELF file SHA256:  2af7743e892cd34e...
* ESP-IDF:         v3.3-163-g601a03e
```

Upgraded OTA Version: 0.9.4

```
12 773 [iot_thread] INFO: NVS> iot_thing_name = [MiniHubPro-3D807C]
13 773 [iot_thread] OTA Version 0.9.4
```

Update the "SUCCEEDED" message to AWS IoT:

```
42 1293 [OTA Agent Task] [prvParseJobDoc] Size of OTA_FileContext_t [64]
43 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ clientToken:
0:MiniHubPro-3D807C ]
44 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ jobId: AFR_OTA-
minihubpro_ota_demo_0001 ]
45 1294 [OTA Agent Task] [prvParseJSONbyModel] Identified parameter [ self_test ]
46 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ updatedBy: 589826 ]
47 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ protocols: ["MQTT"] ]
48 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ streamname: AFR_OTA-
7bd6fc8c-d14f-4a3c-8789-08aee20d6cc4 ]
49 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ filepath: P11_CSK ]
50 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ filesize: 1312384 ]
51 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ fileid: 0 ]
52 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ certfile: P11_CSK ]
53 1294 [OTA Agent Task] [prvParseJSONbyModel] Extracted parameter [ sig-sha256-ecdsa:
MEUCIQC11uQ8fw/5qJbMeVJYGVbvXULR... ]
```

```

54 1294 [OTA Agent Task] [prvParseJobDoc] In self test mode.
W (13260) ota_pal: Set image as testing!
55 1305 [OTA Agent Task] [prvUpdateJobStatus_Mqtt] Msg:
{"status":"IN_PROGRESS","statusDetails":{"self_test":"active","updatedBy":"0x90004"}}
56 1305 [OTA Agent Task] [INFO ][MQTT][13050] (MQTT connection 0x3fff979c) MQTT PUBLISH
operation queued.
57 1306 [OTA Agent Task] [INFO ][MQTT][13060] (MQTT connection 0x3fff979c, PUBLISH
operation 0x3fffc000) Waiting for operation completion.
58 1312 [OTA Agent Task] [INFO ][MQTT][13120] (MQTT connection 0x3fff979c, PUBLISH
operation 0x3fffc000) Wait complete with result SUCCESS.
59 1312 [OTA Agent Task] [prvUpdateJobStatus_Mqtt] 'IN_PROGRESS' to $aws/things/MiniHubPro-
3D807C/jobs/AFR_OTA-minihubpro_ota_demo_0001/update
60 1313 [OTA Agent Task] [prvOTA_Close] Context->0x0x3fffe90
61 1313 [OTA Agent Task] [prvOTAAgentTask] Called handler. Current State [WaitingForJob]
Event [ReceivedJobDocument] New state [CreatingFile]
62 1313 [OTA Agent Task] [prvInSelfTestHandler] prvInSelfTestHandler, platform is in self-
test.
63 1314 [OTA Agent Task] [prvStartSelfTestTimer] Starting OTA_SelfTest timer.
64 1314 [OTA Agent Task] Received eOTA_JobEvent_StartTest callback from OTA Agent.
65 1322 [OTA Agent Task] [prvStopSelfTestTimer] Stopping the self test timer.
66 1322 [OTA Agent Task] [prvUpdateJobStatus_Mqtt] Msg:
{"status":"SUCCEEDED","statusDetails":{"reason":"accepted v0.9.4"}}

```

Go to AWS IoT to check the status. The status is "Succeeded"

The screenshot shows the AWS IoT Jobs console for a job named "AFR_OTA-minihubpro_ota_demo_0001". The job status is "COMPLETED". The overview section shows a summary of job outcomes: 0 Queued, 0 In progress, 0 Timed out, 0 Failed, 1 Succeeded, 0 Rejected, 0 Canceled, and 0 Removed. Below this, a table lists the resources and their status. The resource "MiniHubPro-3D807C" is shown with a status of "Succeeded" and a last updated time of "Jun 16, 2020 8:30:22 PM +0800".

Resource	Last updated	Status
MiniHubPro-3D807C	Jun 16, 2020 8:30:22 PM +0800	Succeeded

*Note:

OTA polling feature will be supported from the firmware **AWS APP Version 0.9.20** or latest so MiniHub Pro doesn't need power on again to trigger the OTA job process.

8 Q&A

8.1 Where is the FW version info?

Please make sure your MiniHub Pro is under the Configuration Mode.
You could find the FW version in the "Configuration AWS & Setting" page.

8.2 Why I cannot see the fw version info?

Your MiniHub Pro firmware version is probably less than 0.9.30. Please upgrade your MiniHub Pro first

8.3 Can I disable the AWS OTA task?

Yes, you could uncheck the "Enable OTA" checkbox in the "Configuration AWS & Setting" page.

8.4 Why I could not find the "Enable OTA" option in the "Configuration AWS & Setting" page?

Your MiniHub Pro firmware version is probably less than 0.9.38. Before v0.9.38, the AWS OTA task is necessary. If you want to disable the AWS OTA task, please upgrade your MiniHub Pro first