aws

Onboarding-by-Claim Customer/OEM Guide

AWS IoT ExpressLink

AWS IoT ExpressLink: Onboarding-by-Claim Customer/OEM Guide

Table of Contents

AWS IoT ExpressLink Onboarding-by-Claim Customer/OEM Guide

AWS IoT ExpressLink modules are hardware modules that enable easy cloud connectivity and security for AWS IoT devices. OEMs can integrate ExpressLink modules into their products to accelerate IoT development. ExpressLink modules come pre-provisioned with unique credentials to authenticate with AWS IoT Core. There are several options for onboarding ExpressLink devices, including individual/batch uploads, just-in-time provisioning, and fleet provisioning. The description focuses on a novel "onboarding-by-claim" process specific to ExpressLink. This leverages the module's unique capabilities to provide a more secure and streamlined onboarding experience.

Overview

AWS IoT ExpressLink modules are hardware connectivity modules that enable easy AWS cloud connectivity and implement strict and AWS-mandated security requirements for device-to-cloud connections. OEMs can accelerate the development of IoT products by integrating ExpressLink modules into their designs.

ExpressLink modules come pre-provisioned with a unique identifier and a certificate signed by the module manufacturer's Certificate Authority (CA), ready to authenticate with AWS IoT Core. *Onboarding* refers to the act of binding the module's credentials to a `thing` inside the AWS IoT registry of an OEM's account. There are multiple ways to onboard devices:

- individual certificate upload
- batch certificate upload
- just-in-time provisioning (JITP)
- just-in-time registration (JITR)
- fleet provisioning

The onboarding process

This guide describes a novel *onboarding-by-claim* mechanism specifically created to leverage an ExpressLink module's unique capabilities.

By default, ExpressLink modules connect to ExpressLink *staging account endpoints*. The staging account is managed by AWS to facilitate the onboarding-by-claim mechanism. It acts as a spring board to dispatch devices to their ultimate destination– the customer/OEM account. (See the AWS IoT ExpressLink Getting Started Guide).

The onboarding-by-claim process uses the Just-in-time provisioning (JITP) mechanism to automatically upload the device certificate, associate a policy, and create a "thing", but provides additional features, including:

- Late binding– the onboarding happens only when the end-user activates a finished product. This makes the onboarding process less time consuming during product manufacturing and, therefore, less expensive.
- No disclosure of confidential information is required with any element of the supply chain. This makes the process more secure and flexible, as the supply chain of trust is reduced to a direct link from the ExpressLink module manufacturer to the end-user in possession of the finished product.

The Onboarding Process in Detail: the user experience

1. The onboarding-by-claim process is driven by the end-user who purchased a finished IoT capable product. It is triggered when the user interacts with a "product registration portal". This is OEM-specific software that can be a web or mobile application that offers end-users the opportunity to bind their identity to the unique product in their possession. After that, the OEM's application(s) can offer unique, personalized services augmented by the product's IoT connectivity.
2. During product registration, the user is instructed to turn on the product and connect it for the first time. If the product uses an ExpressLink module, it connects to the default staging account, unless otherwise configured. Then, it automatically subscribes to a configuration (MQTT) topic.
3. The end-user is instructed to enter a unique identifier that they can find on a label on the finished product or its packaging. This identifier is a long alphanumerical string or possibly a QR code (the preferred option if the registration portal is implemented as a mobile app).
4. The identifier can now be used to find the unique "thing" present in the staging account and communicate with it (using the configuration topic) to provide the desired customer/OEM endpoint. It is this latter action that constitutes the actual "claim". After that, the device disconnects from the staging account, and then connects to the newly assigned endpoint.

5. The just-in-time provisioning (JITP) mechanism completes the process– the new device is authenticated, and a new "thing" is created in the selected customer/OEM account according to the instructions provided by a JITP template.

Enable onboarding-by-claim

In order for a product to take advantage of the onboarding-by-claim process, the OEM must ensure that the following components are available and properly configured:

- A registration (web) portal or mobile application.
- The claim-script - this sends the new endpoint to the selected device inside the staging account.
- The customer/OEM account - this must be properly configured to support just-in-time provisioning.

The following sections explain how these components operates, and describe how to properly configure, and provide implementation examples.

Create a registration application

The registration application is a software product that is (ideally) completely customized for the specific OEM product and brand. It requests the end-user to input the device's unique identifier, and then launches the claim-script that provides the device with a new target endpoint.

Note that the registration portal can be responsible for collecting additional end-user information, such as the end-user's location, and for including personally identifiable information that can be used by the application's location to select a target endpoint among several alterantives (if the OEM controls multiple AWS accounts). This also makes it possible to optimize the end-user experience and reduce latency (by selecting the most appopriate region, for example). The logic used, and any additional information optionally collected, are completely outside the scope of this document and are not essential to the onboarding process.

For an example implementation of a basic (web) registration portal, refer to the claim provisioning reference implementation (download).

Claim implementation

The actual claim is implemented in the claim-script (a function of the registration application). To do this, it publishes a specific, JSON formatted message on the unique device configuration topic. This operation requires the script to obtain access to the ExpressLink module staging account which is managed by AWS. While AWS does not share such credentials with customers/OEMs, upon request the AWS IoT Device service team allows the customer/OEM to create a "claim-thing" within the staging account registry. The claim-thing can then be controlled by the OEM registration application (using an MQTT client API) to publish the endpoint update message. To request and obtain control of a claim-thing follow the configuration steps indicated in Appendix A - Steps to obtain a claim-thing.

Configure the OEM account

In the last step, the customer/OEM's AWS account must be configured to enable the use of the just-in-time provisioning mechanism. To do this:

1. Register the ExpressLink module vendor's Certificate Authority with the customer/OEM account. Follow the steps in Appendix B - Register the ExpressLink manufacturer certificate authority (CA).
2. Create a JITP template so that new devices that are directed to the account will be automatically associated with a desired policy and given a proper thing-name. Follow the steps in Appendix C - Create a JITP template.

Summary

To summarize, ExpressLink modules come pre-provisioned with a unique identifier and a certificate signed by the module manufacturer Certificate Authority (CA), ready to authenticate with AWS IoT Core.

Onboarding, the act of binding the module credentials to a thing inside the AWS IoT registry of an customer/OEM's account is accomplished using various mechanisms provided to all devices that connect to AWS IoT Core. This guide describes a novel onboarding-by-claim mechanism specifically created to leverage an ExpressLink module's unique capabilities.

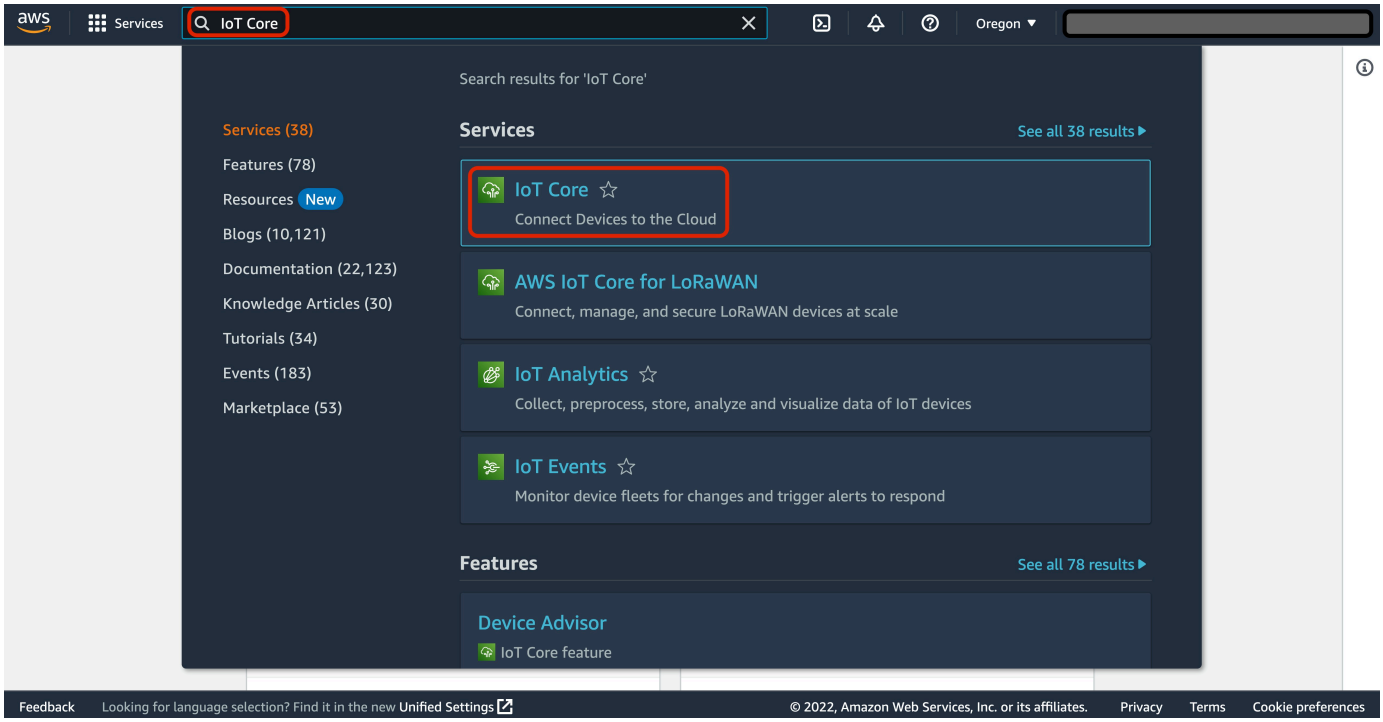
By following the steps in this document, any customer/OEM can take advantage of this new capability to provide their own customers with the best experience, while optimizing the supply chain for security and flexibility.

Appendix A - Steps to obtain a claim-thing

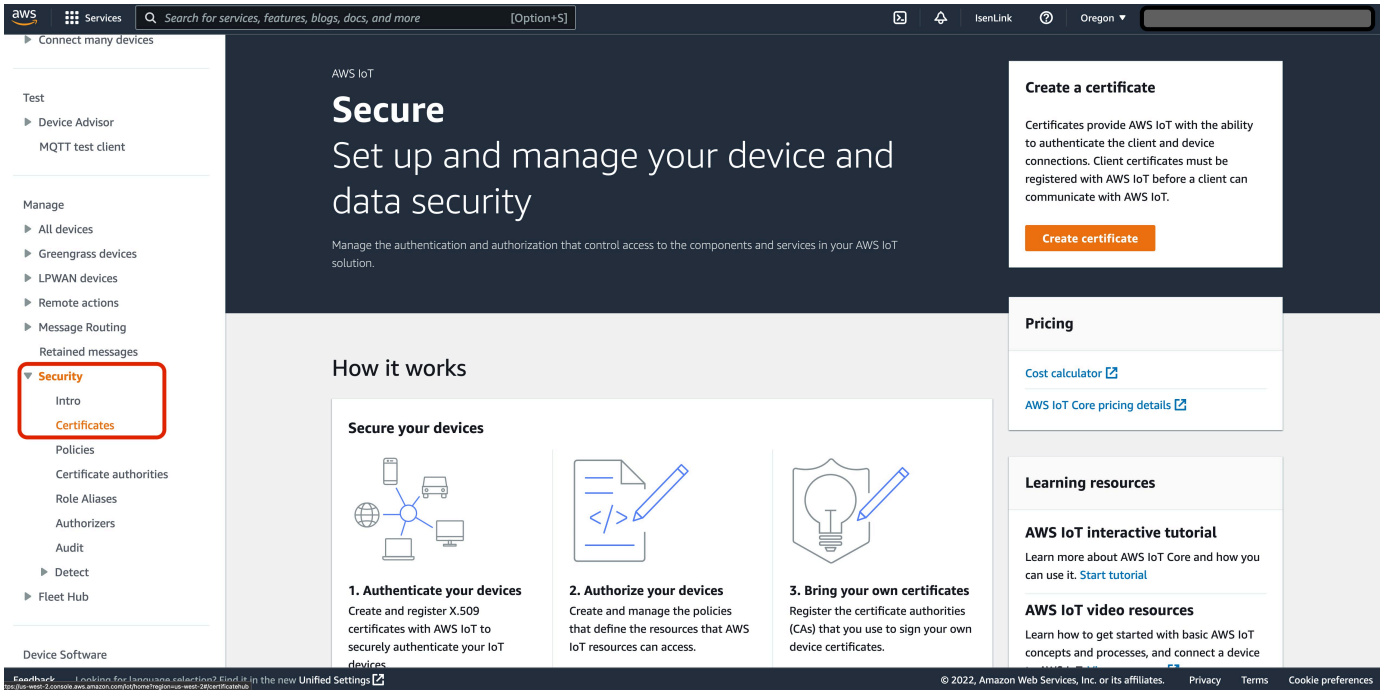
This appendix talks about the steps to obtain a claim-thing. To generate a claim-thing certificate, follow this 7 step process. Once the certificate is generated, the next thing to do is to register as an OEM with AWS. This is a 3 step process which details information about the registration.

Generate a "claim-thing" certificate

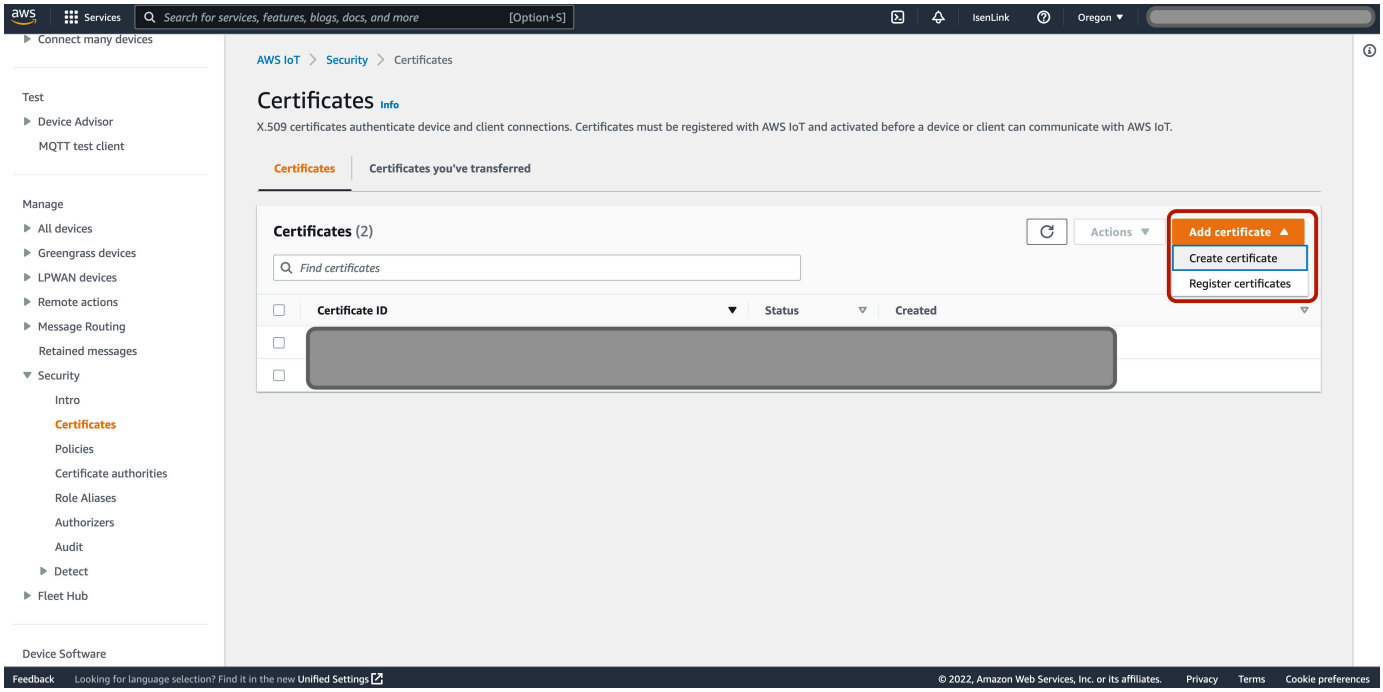
1. If you do not have AWS account, or wish to use a new one specifically for ExpressLink follow the steps to Create an AWS account.
2. If you aren't already signed in to your AWS account, sign in, then open the AWS IoT console.



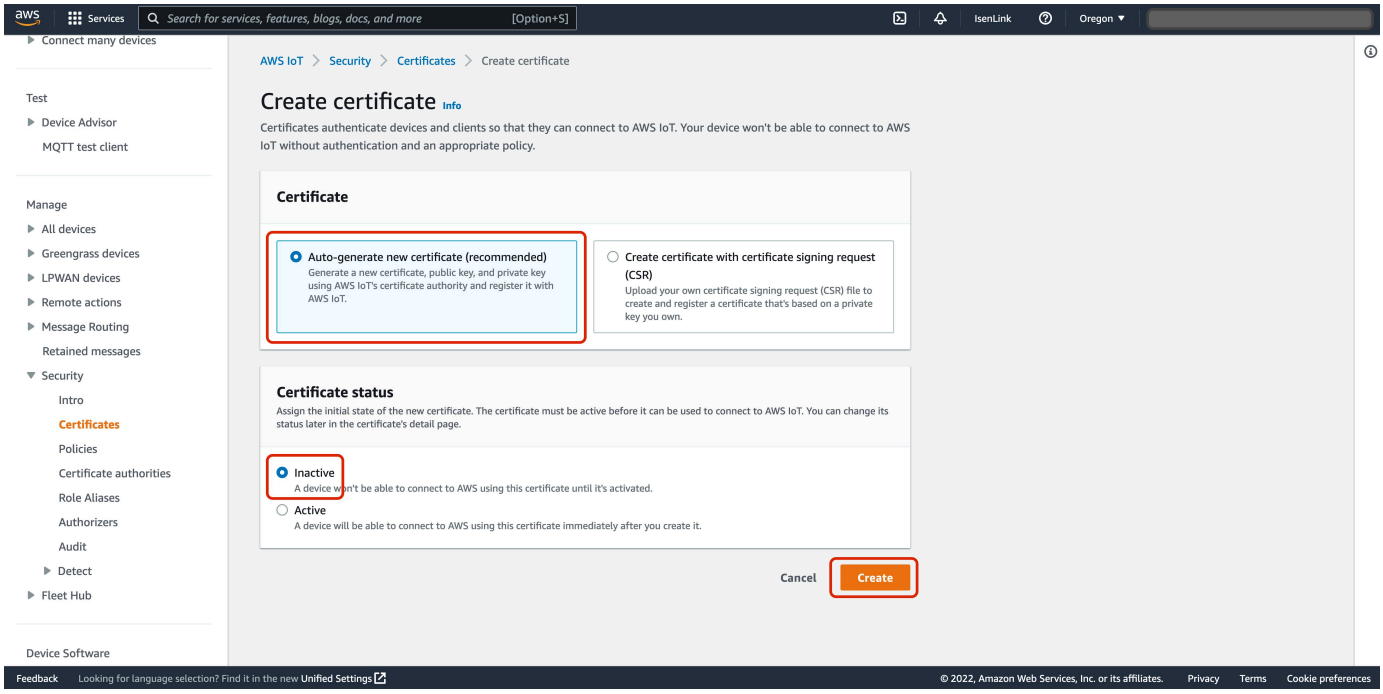
3. In the AWS IoT console, on the left navigation pane, select **Security** to expand the sub-menu, then select **Certificates**.



4. On the **Certificates** page, on the right side of the table that shows currently-installed certificates, select **Add certificate**, then select **Create certificate** in the drop-down menu.



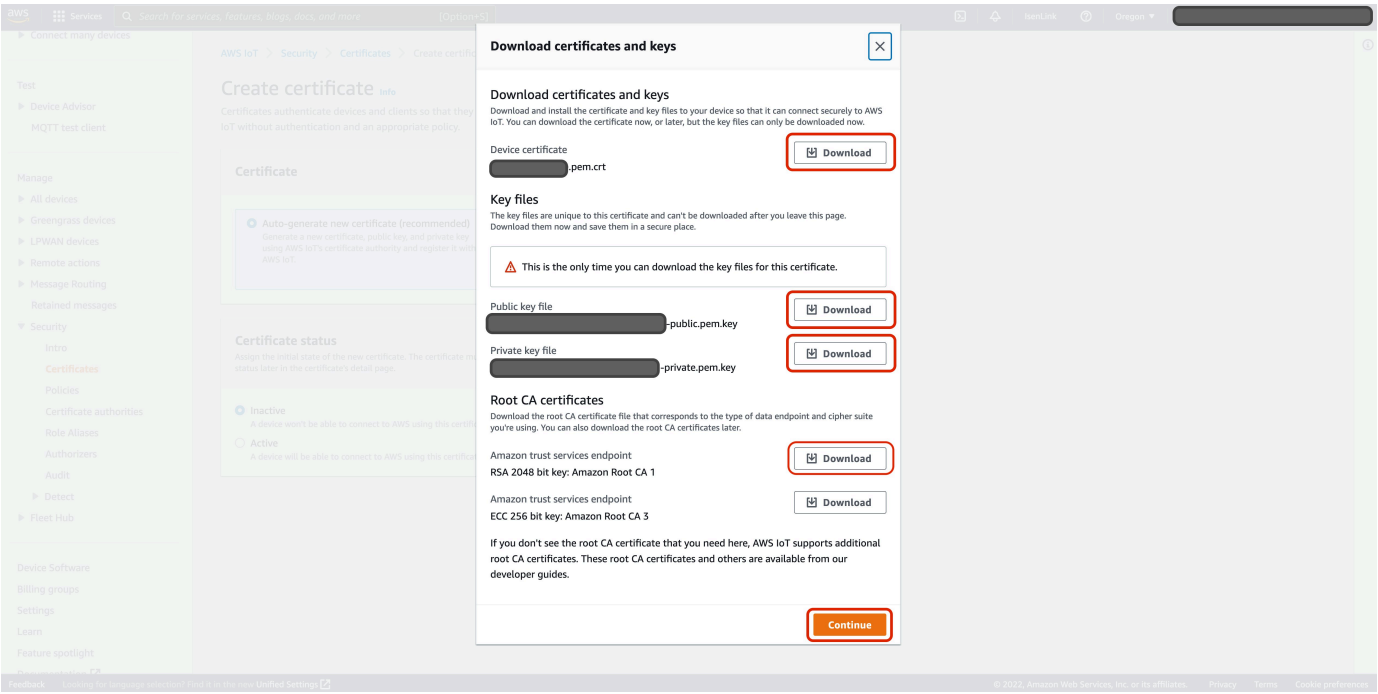
5. On the **Create certificate** page, choose **Auto-generate new certificate**, and choose **Inactive**. Select **Create** to create an X.509 certificate.



6. In the pop-up window that opens, select **Download** for each of the credentials files that you will need:

   - *certificate fingerprint*.pem.crt
   - *certificate fingerprint*-public.pem.key
   - *certificate fingerprint*-private.pem.key
   - Amazon Root CA 1 (this file will be downloaded as AmazonRootCA1.pem).

   (The *certificate fingerprint* is a hexadecimal string that uniquely identifies the certificate and is generated using the certificate body.)



7. Select **Continue** to close the pop-up window, then store the keys and the certificate in a safe place following security best practices.

Register as an OEM with AWS

1. Send an email with the following information to <expresslink-onboarding@amazon.com>:
   - Company name
   - AWS account ID
   - Technical/Developer Contact (name and email)
   - Technical Manager Contact (name and email)
2. When it receives the request, the AWS IoT ExpressLink service team will:
   - provide a secure mechanism for you to upload the certificate generated in the previous section.
   - create a universally unique identifier (UUID), a 128-bit string label for your onboarding functionality. The UUID is required to connect to the Staging Endpoint.

The AWS IoT ExpressLink service team will send the UUID for the onboarding functionality, instructions for uploading the certificate, related documentation, and terms & conditions to the two technical contacts listed in your request.

3. After you receive the information listed in the previous step, follow the instructions and upload the certificate (*`certificate fingerprint`*`.pem.crt`) that you generated in the previous section.

> **Warning**
>
> DO NOT upload the private key! (*`certificate fingerprint`*`-private.pem.key`).

Appendix B - Register the ExpressLink manufacturer certificate authority (CA)

After a claim-thing certificate is obtained, the next steps is to register the ExpressLink certificate with the manufacturer certificate authority (CA). This four step process walks the user through registering with the certificate authority (CA).

1. Follow the steps in Getting started with the AWS CLI to install the AWS CLI on your development machine.
2. Follow the steps in Configuration and credential file settings to configure the AWS CLI to use your AWS account credentials.
3. Register the root CA on your AWS account with the following AWS CLI command (replace *`path-to-manufacturer-CA`* with the local path of the the root CA):

```
aws iot register-ca-certificate --ca-certificate file://path-to-manufacturer-CA --
certificate-mode SNI_ONLY --set-as-active --allow-auto-registration
```

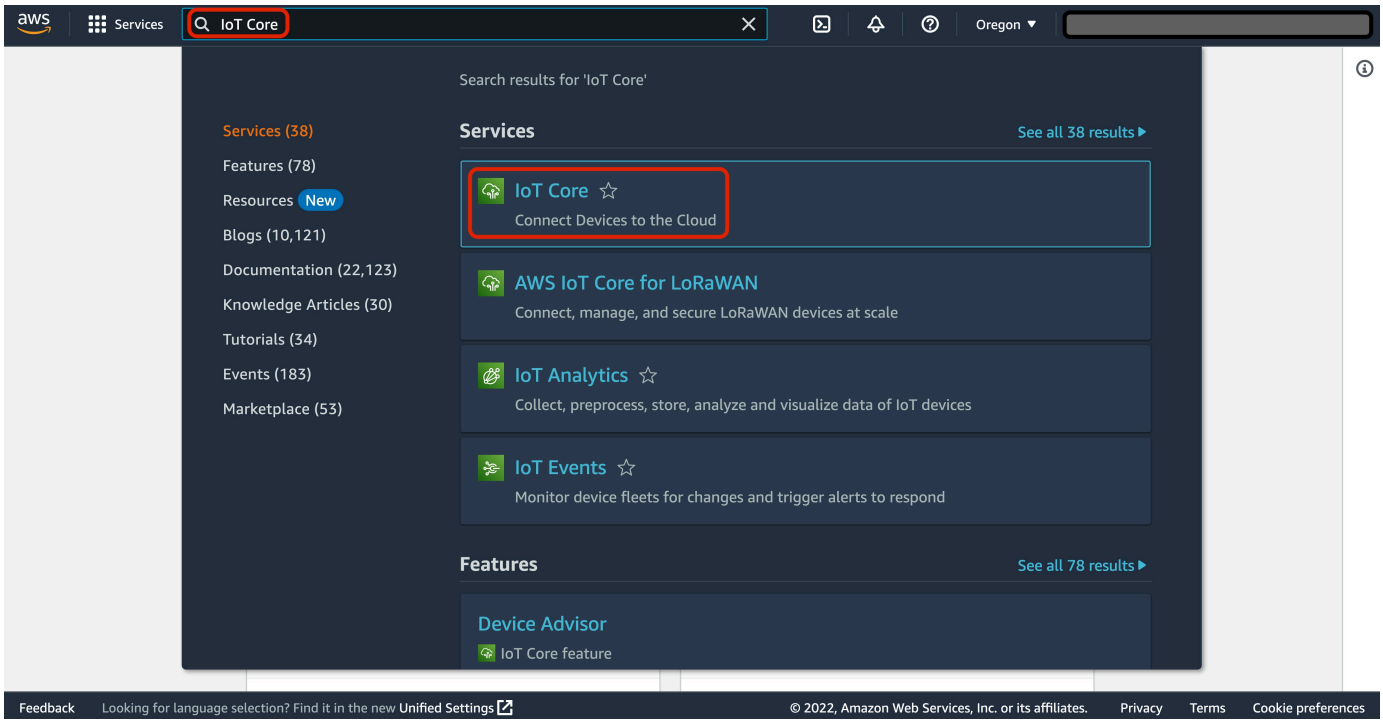
(For more information on non-Amazon-signed certificates and certificate authorities on AWS IoT Core, see Create your own client certificates.

4. Record the *CA certificate id* that is shown in the output of the command above. The CA certificate id is a long hexadecimal string. You will need this later.

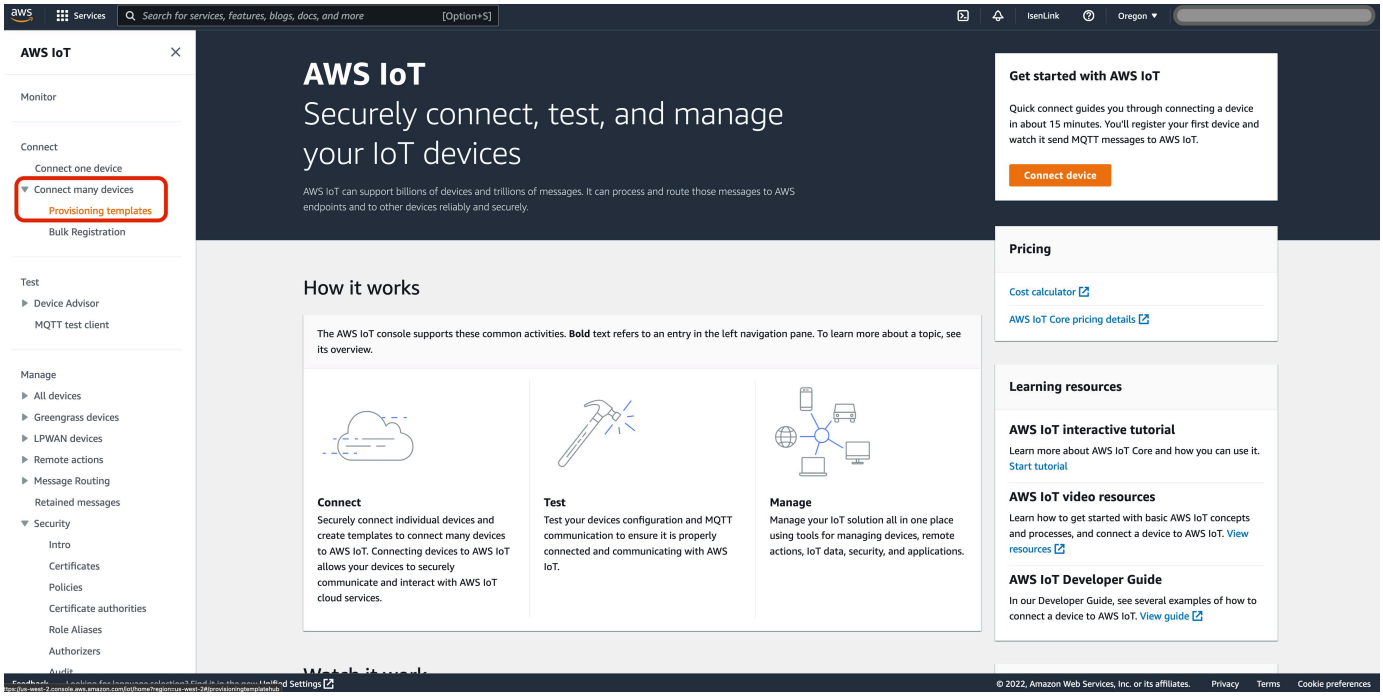
Appendix C - Create a JITP template

Create a JITP template so that new devices directed to the account are automatically associated with a desired policy, and given a proper thing-name on the AWS console. Follow the steps below to create a template.

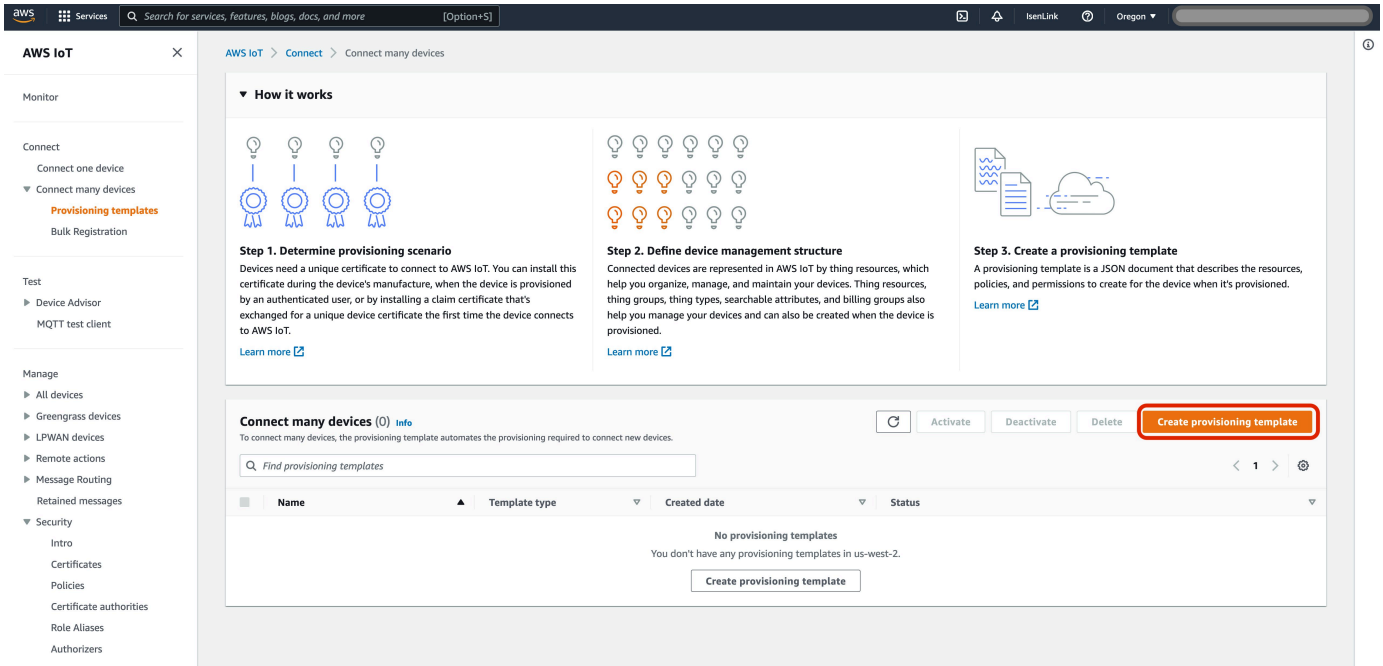
1. Open the AWS IoT console.



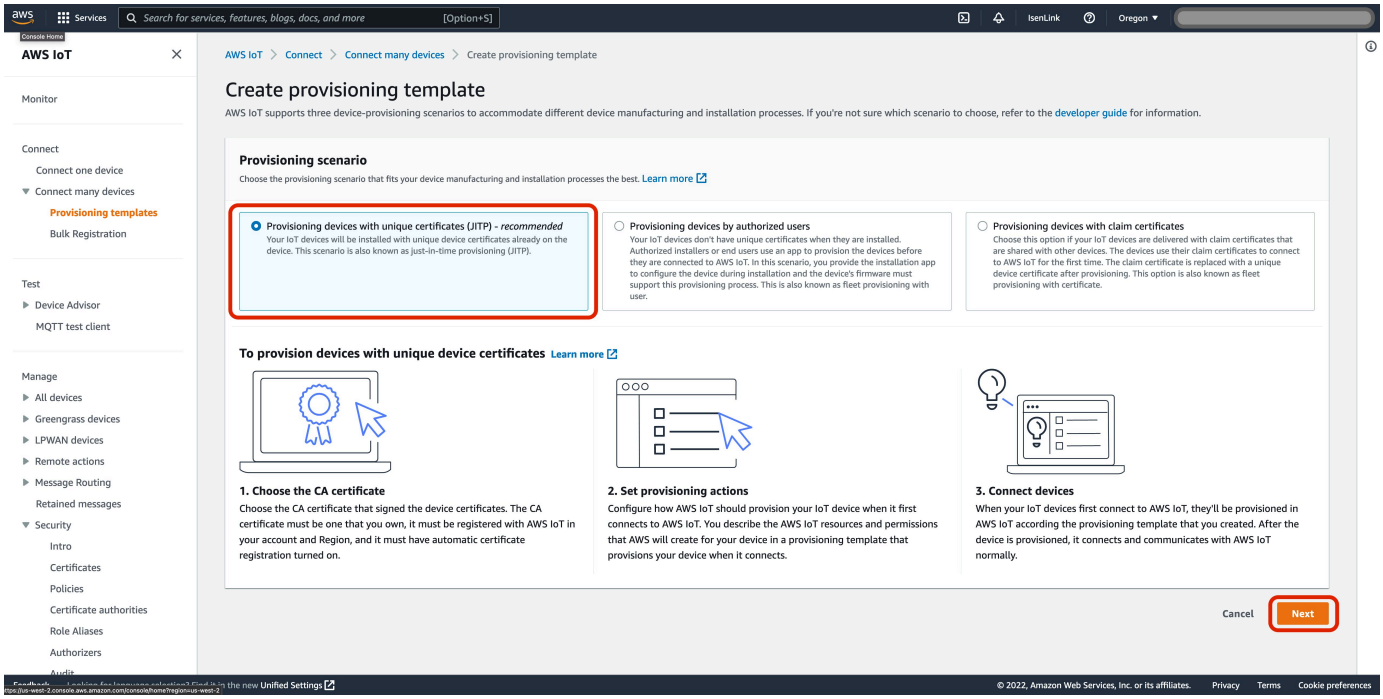
2. In the left navigation pane, select **Connect many devices** then select **Provisioning templates** in the drop-down sub-menu.



3. On the provisioning templates management page, select **Create provisioning template**.



4. On the **Create provisioning template** page, choose **Provisioning devices with unique certificate (JITP) - *recommended***, then select **Next**.

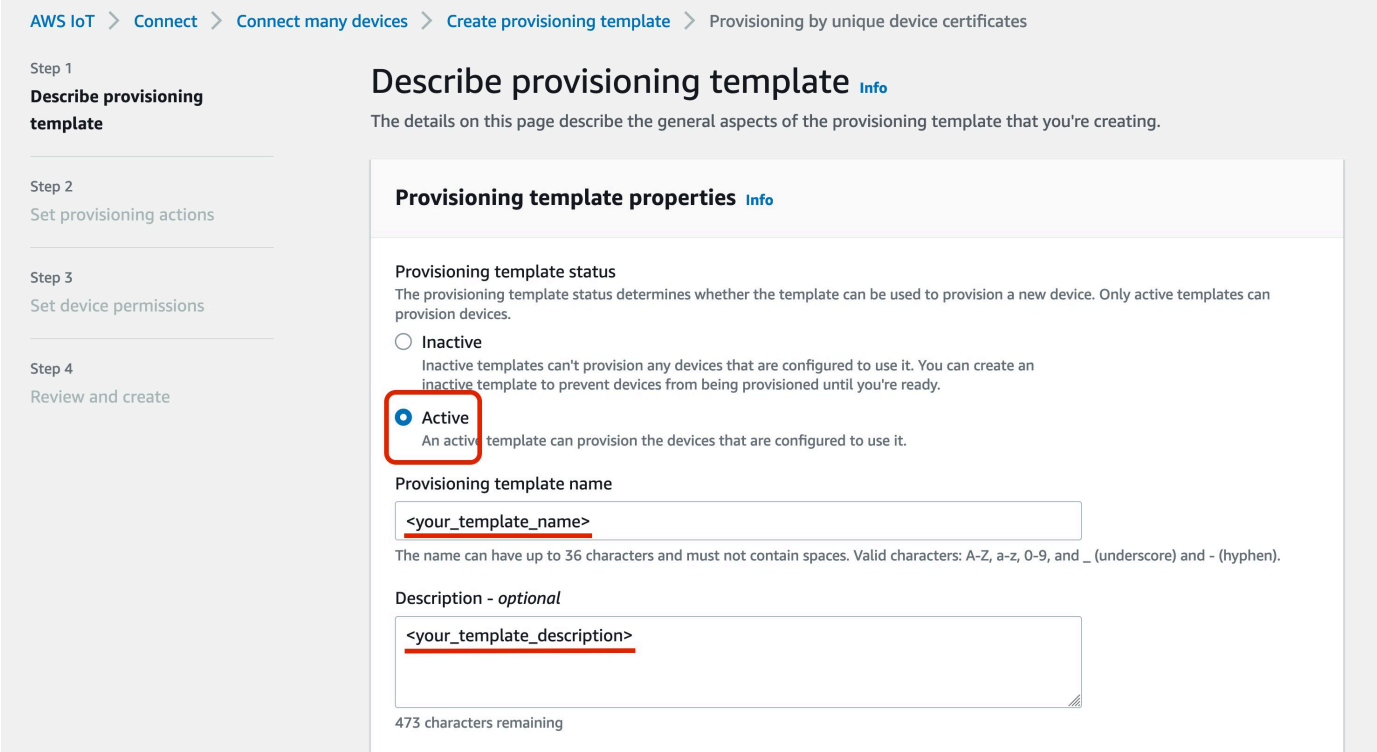


5. In the JITP template creation wizard, under **Describe provisioning template**, enter the information for the **Provisioning template properties**:

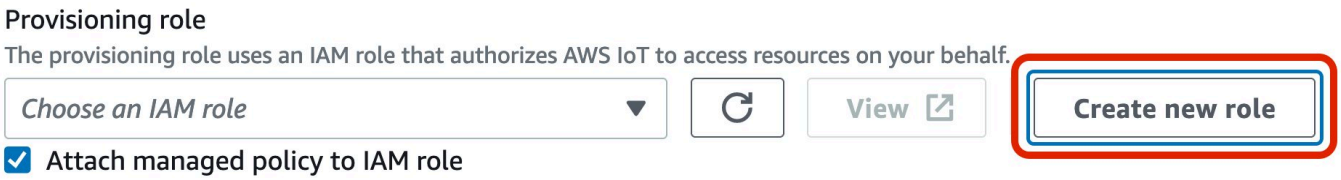
   a. Under **Provisioning template status**, choose **Active**.

   b. Enter a **Provisioning template name**.

   c. (Optional) Enter a **Description** for the template.

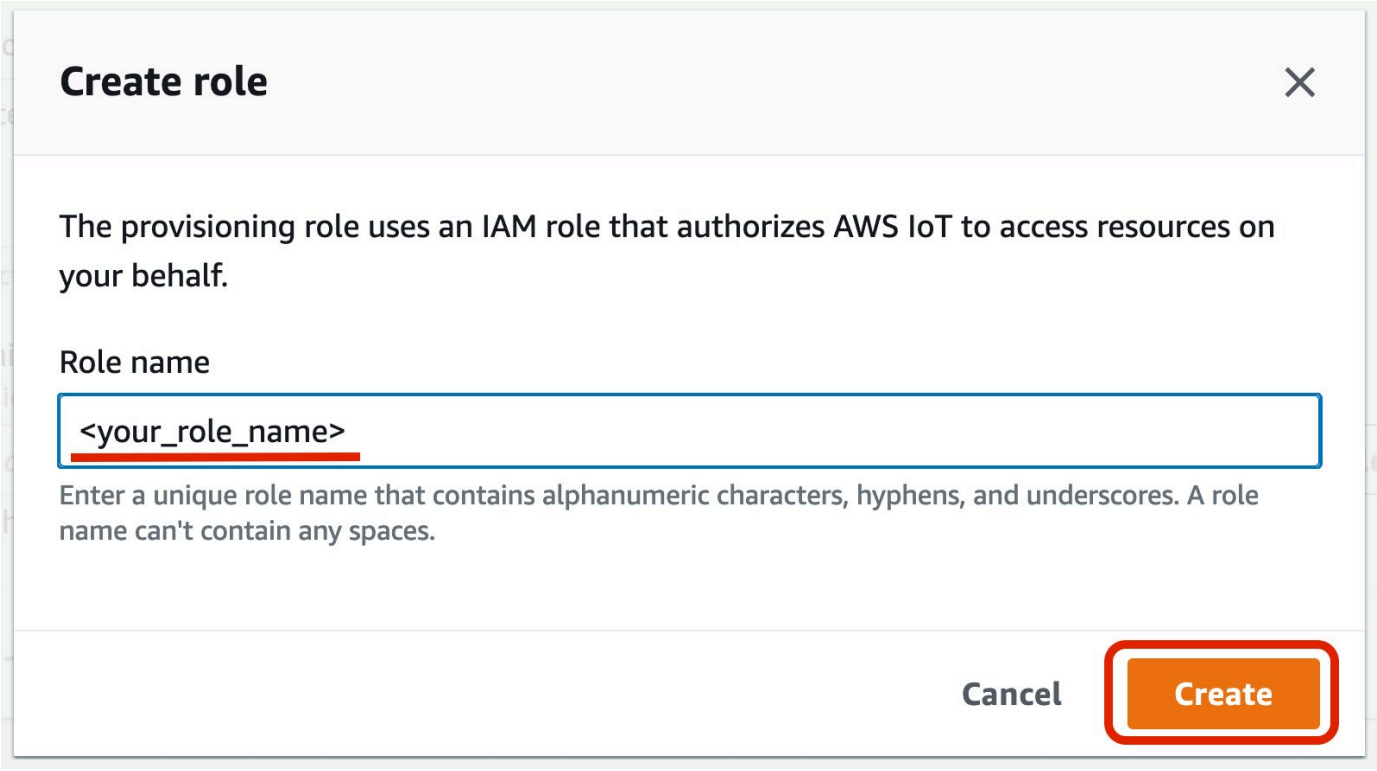


6. Under **Provisioning role**, make sure **Attach managed policy to IAM role** is checked. (This ensures the IAM role created here and used in device provisioning will have the needed privileges.) Then select **Create new role**.

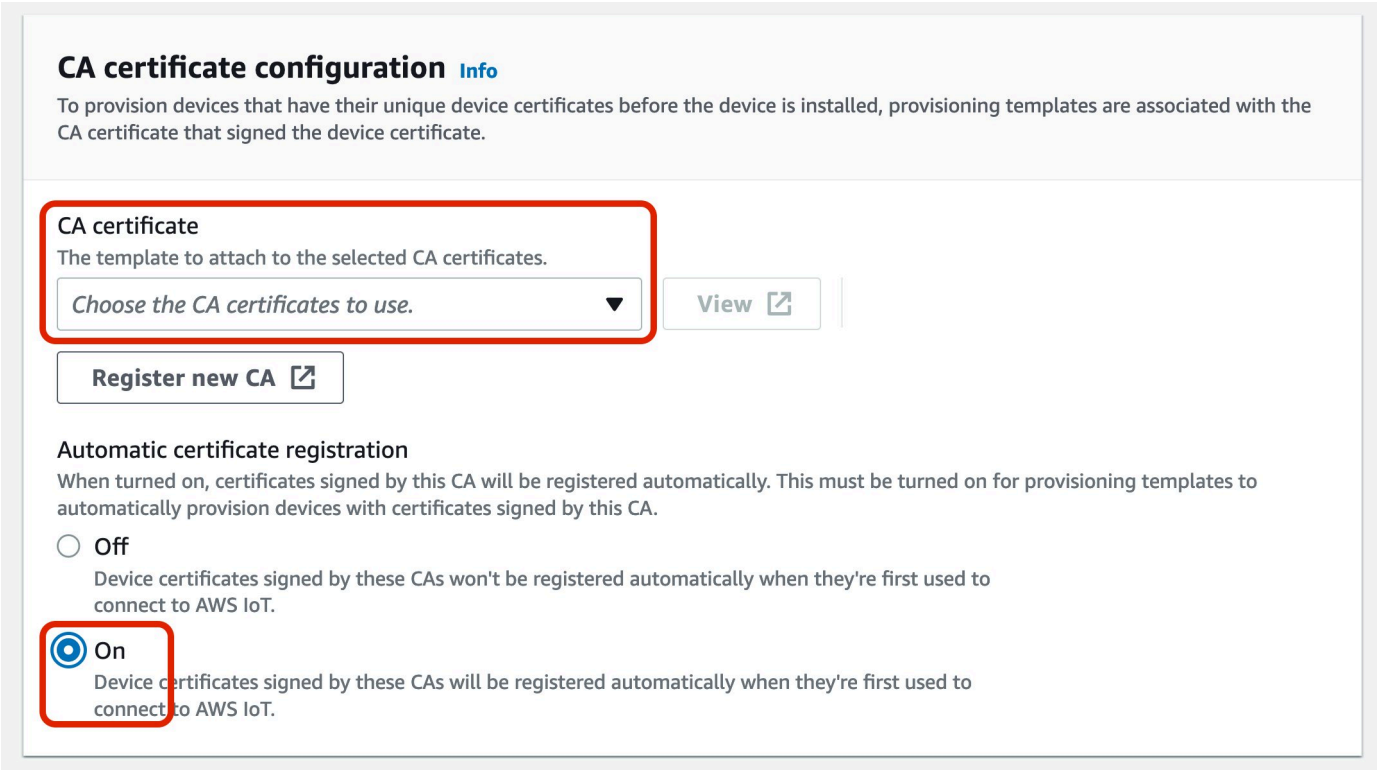


(Optional) Instead of creating a new role, you can choose a role that you have previously made. However, to make sure that the role has enough privileges to provision your ExpressLink modules, you must make sure that the role has the AWS managed policies "`AWSIoTThingsRegistration`", "`AWSIoTLogging`", and "`AWSIoTRuleActions`" attached, or that it has an inline policy with equivalent or greater permissions. See AWS managed policies for AWS IoT for more information.

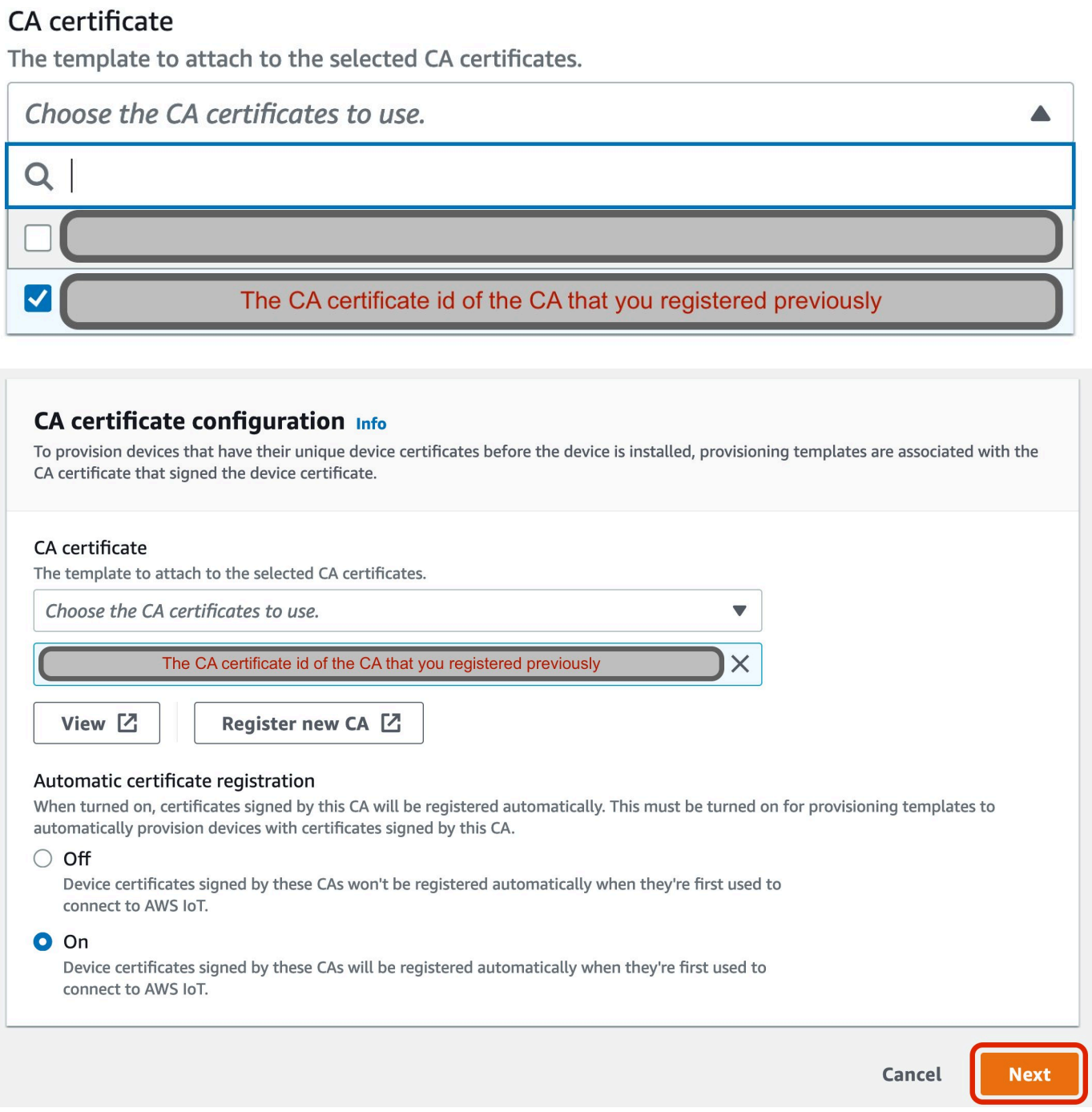
7. In the **Create role** pop-up window, enter a **Role name**, then select **Create**.



8. Under **CA certificate configuration**, for **Automatic certificate registration**, choose **On**. Above that, under **CA certificate** select the **Choose the CA certificates to use** dropdown menu.



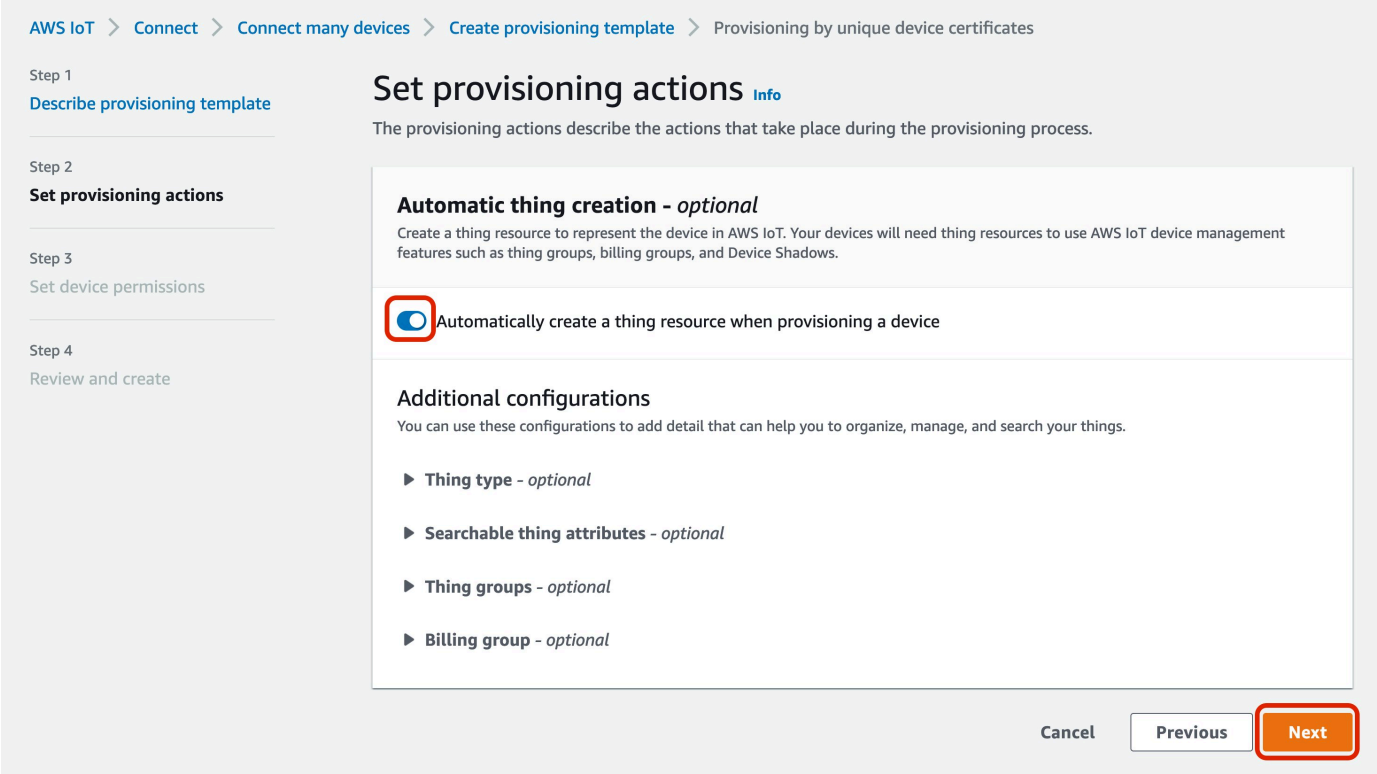
9. In the dropdown menu, choose the checkbox in front of the CA certificate ID that was listed in the output of the `aws iot register-ca-certificate` AWS CLI command that you ran in the previous section. After the CA certificate ID appears on the page, select **Next**.



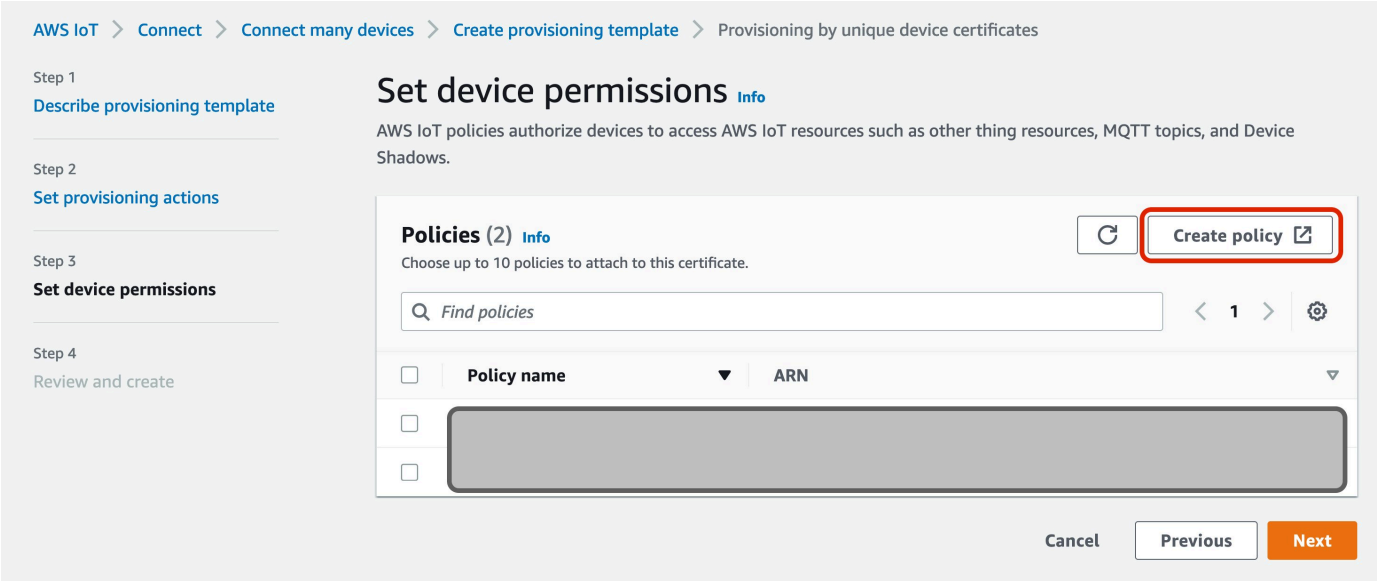
10. Under **Set provisioning actions**, toggle on **Automatically create a thing resource when provisioning a device**.

(Optional) You can also choose **Additional configurations** for the Thing type, Searchable thing attributes, Thing groups, and Billing groups.

Select **Next**.

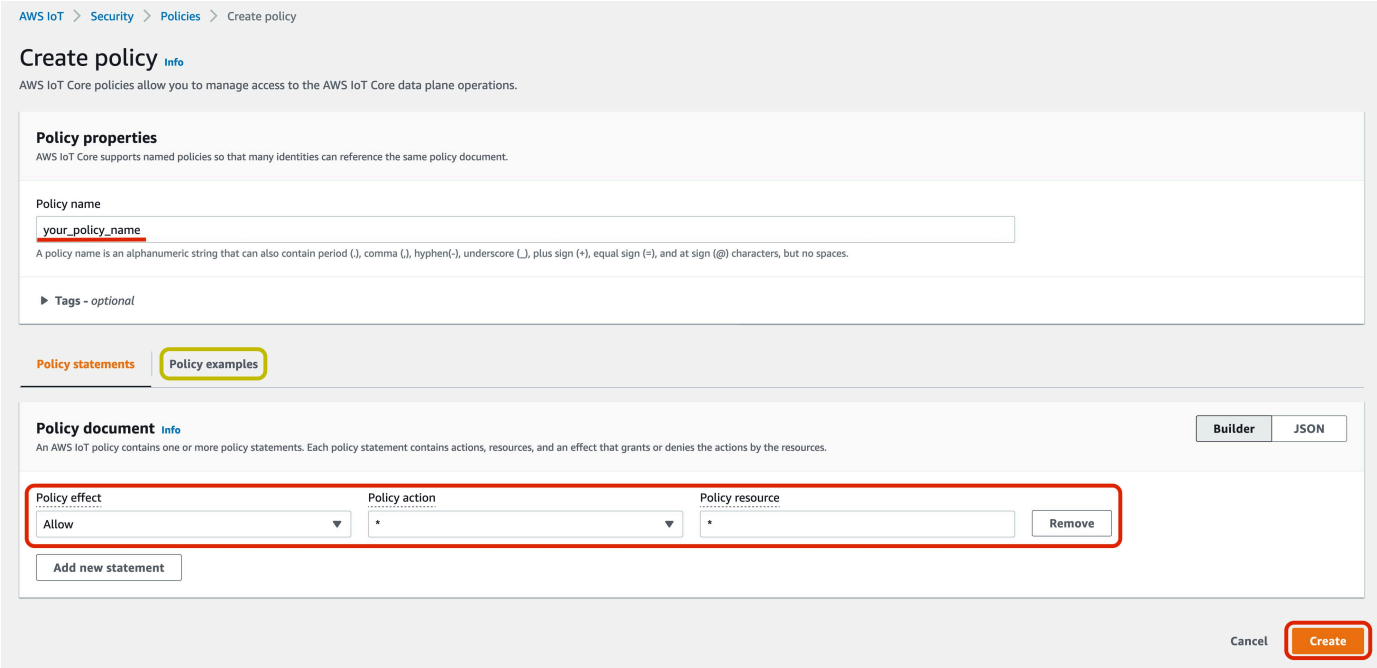


11. Under **Set device permissions**, select **Create policy** to create a new policy for those ExpressLink modules you want to provision. This policy determines the actions that can be run by the ExpressLink modules on AWS IoT Core under your AWS account.



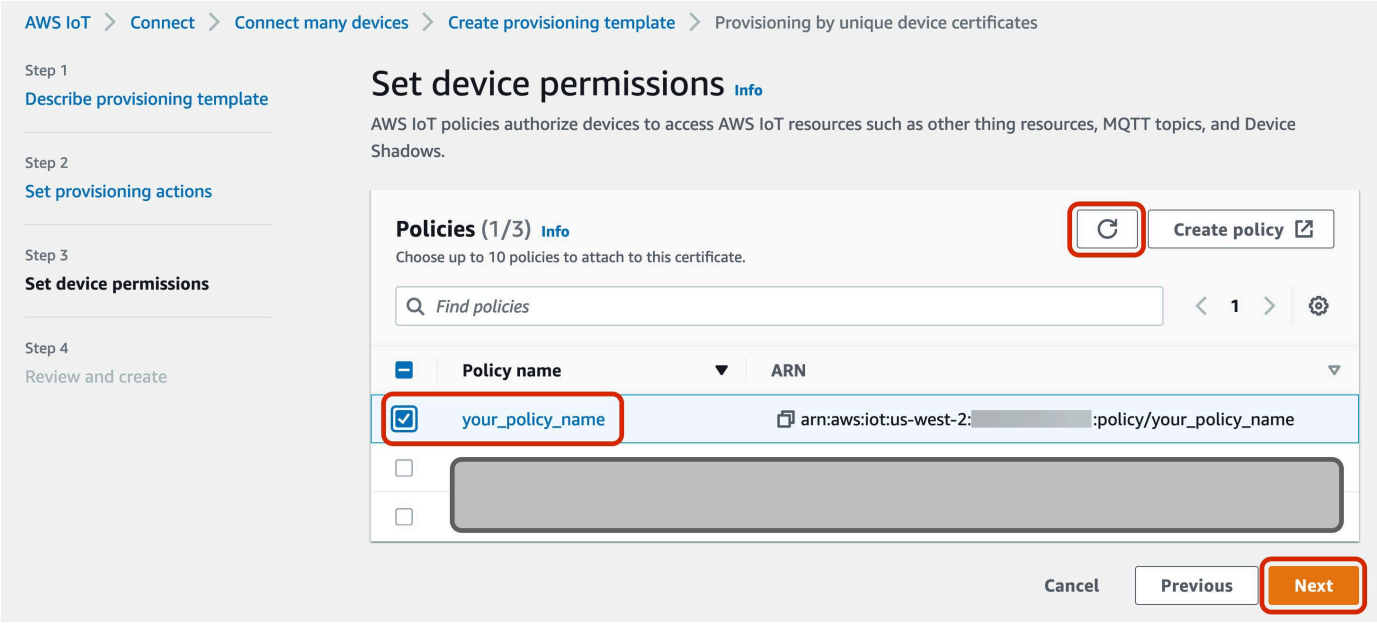
12. On the **Create policy** page, enter a **Policy name**.

    On the **Policy statements** tab, under **Policy document**, for convenience you can enter "*" for both the **Policy action** and **Policy resource**. However, this policy will allow any and all actions on any AWS IoT Core resources accessible through MQTT. We recommend that you use a more restrictive policy. The **Policy examples** tab contains numerous example policy documents that can be applied for different use cases. See the policy documents under that tab or refer to AWS IoT Core policies for additional information.

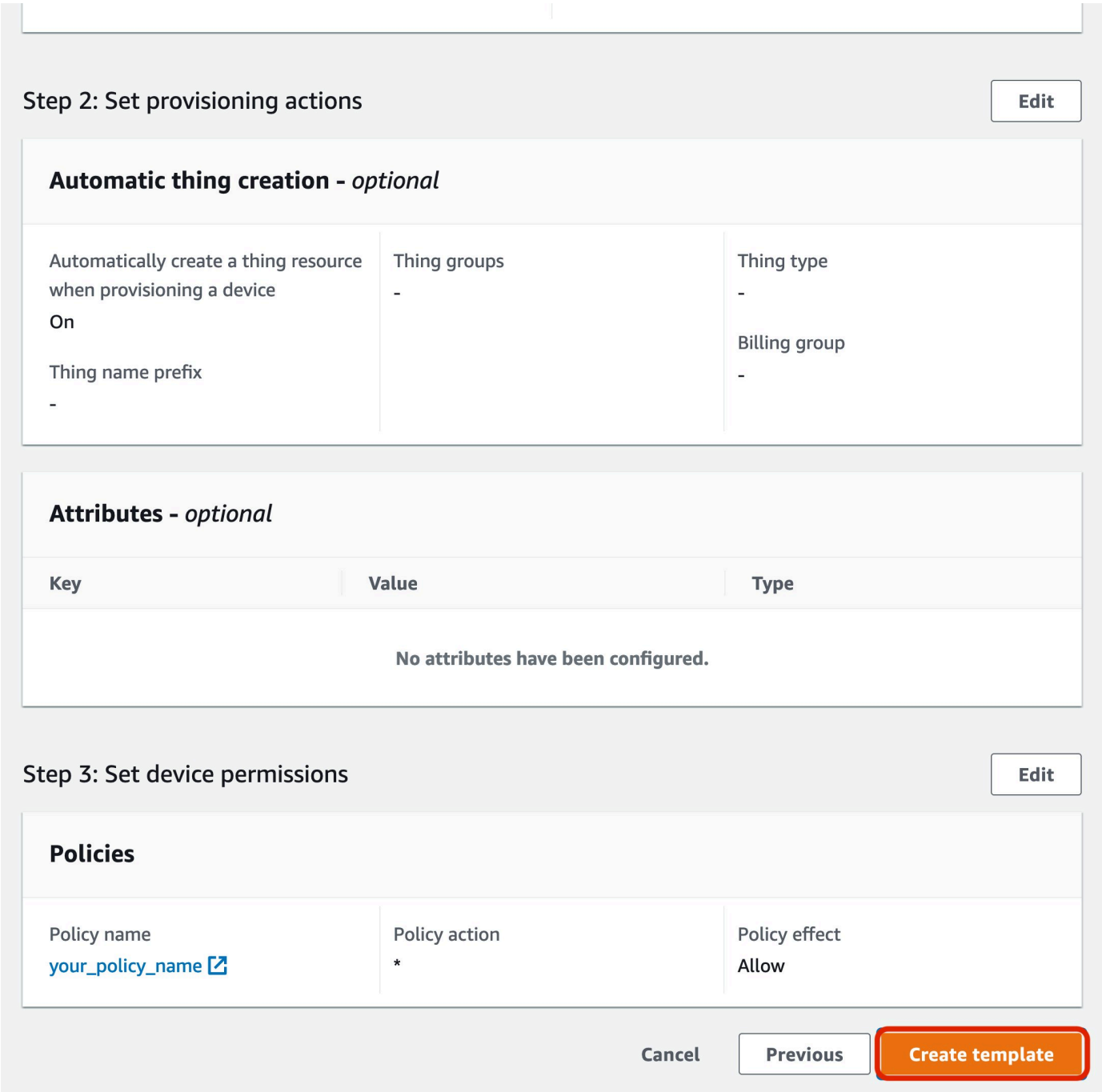


13. Select **Create** to create the policy.

14. On the **Set device permissions** page, refresh your browser (select the icon that looks like a circular arrow pointing to its own starting point). Your policy should now show up in the list under **Policies**. Choose the checkbox in front of your policy, then select **Next**.



15. Review the information you have entered to make sure it is correct. In particular, make sure that the CA certificate shown under **CA certificate configuration** has the same CA certificate ID returned by the AWS CLI command that you ran in a previous step. You can edit a particular section if the information is incorrect. After you verify the information you entered, scroll to the bottom and select **Create template**.



16. Your JITP template is now created and ready to be applied whenever an ExpressLink module issues a connect request to your AWS account's IoT Core endpoint during onboarding-by-claim.

Appendix D - Glossary

This is a glossary, containing all the terms used in this guide.

ExpressLink modules

Hardware connectivity modules that enable easy AWS cloud connectivity and implement strict and AWS-mandated security requirements for device-to-cloud connections.

ExpressLink devices

OEM products that embed an ExpressLink module and use the module for AWS cloud connectivity.

OEM AWS account

An AWS account that OEMs use to manage their IoT devices.

ExpressLink AWS staging account/endpoint

An AWS-managed account that provides:

- out-of-the-box Quick Connect functionality.
- a staging area for non-onboarded ExpressLink devices waiting to be claimed and onboarded.

onboarding

The registering of an OEM's ExpressLink device inside the OEM AWS Account; ExpressLink provides 5 ways to do this.

onboarding-by-claim

One of the 5 onboarding methods provided by ExpressLink; it streamlines onboarding and eliminates manual setup.

claim-thing

A virtual device in the staging account (registry) whose sole purpose is to publish configuration messages.

registration portal

A web or mobile application that allows the end-user to register an (IoT) product.

Onboarding functionality

Any piece of code that is able to:

- connect to the ExpressLink AWS staging account as the claim-thing.
- publish the MQTT endpoint change message to the MQTT control topic in the staging account.