Quickstart

This page shows you how to use Google Cloud Console to create a Cloud IoT Core device registry and register a device. It also shows you how to run a sample to connect a device and publish device telemetry events.

You can also try out an interactive tutorial (https://console.cloud.google.com/start?tutorial=iot_core_quickstart) in Cloud Console.

Before you begin

1. No Console do Google Cloud, na página do seletor de projetos, selecione ou crie um projeto do Google Cloud.

★ **Observação**: se você não pretende manter os recursos criados neste procedimento, crie um projeto novo em vez de selecionar um que já existe. Depois de concluir essas etapas, é possível excluir o projeto. Para fazer isso, basta remover todos os recursos associados a ele.

Acessar a página do seletor de projetos (https://console.cloud.google.com/projectselector2/hon

2. Verifique se o faturamento está ativado para seu projeto na nuvem. Saiba como confirmar se o faturamento está ativado para o projeto (/billing/docs/how-to/modify-project).
3. Ative as APIs Cloud IoT Core and Cloud Pub/Sub.

Ative as APIs (https://console.cloud.google.com/flows/enableapi?apiid=cloudiot.googleapis.com,

Set up your local environment and install prerequisites

1. Install and initialize the Cloud SDK (https://cloud.google.com/sdk/docs/). Cloud IoT Core requires version 173.0.0 or higher of the SDK.
2. Set up a Node.js development environment (https://cloud.google.com/nodejs/docs/setup).

   Alternatively, you can use Google Cloud Shell (https://cloud.google.com/shell/docs/starting-cloud-shell), which comes with Cloud SDK and Node.js already installed.

Create a device registry

1. Go to the Google Cloud IoT Core page in Cloud Console.

   Go to the Google Cloud IoT Core page (https://console.cloud.google.com/iot)

2. Click **Create registry.**
3. Enter `my-registry` for the **Registry ID**.
4. If you're in the US, select **us-central1** for the **Region**. If you're outside the US, select your preferred region (https://cloud.google.com/iot/docs/requirements#cloud_regions).
5. Select **MQTT** for the **Protocol**.
6. In the **Default telemetry topic** dropdown list, select **Create a topic.**
7. In the **Create a topic** dialog, enter `my-device-events` in the **Name** field.
8. Click **Create** in the **Create a topic** dialog.
9. The **Device state topic** and **Certificate value** fields are optional, so leave them blank.
10. Click **Create** on the Cloud IoT Core page.

You've just created a device registry with a Cloud Pub/Sub topic for publishing device telemetry events.

Create your credentials

Generate a device key pair

Open a terminal window and run the following multi-line command to create an RS256 key:

```
openssl req -x509 -newkey rsa:2048 -keyout rsa_private.pem -nodes \
    -out rsa_cert.pem -subj "/CN=unused"
```

Download root credentials

Download Google's CA root certificate (https://pki.goog/roots.pem) and note the location where you downloaded it. You'll need the file path when you run the Node.js command below.

In the following section, you'll add a device to the registry and associate the public key with the device.

Add a device to the registry

1. On the **Registries** page, select `my-registry`.
2. Select the **Devices** tab and click **Create a device**.
3. Enter `my-device` for the **Device ID**.
4. Select **Allow** for **Device communication**.
5. Add the public key information to the **Authentication** fields.
    - Copy the contents of `rsa_cert.pem` to the clipboard. Make sure to include the lines that say `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----`.
    - Select **RS256_X509** for the **Public key format**.
    - Paste the public key in the **Public key value** box.
    - Click **Add** to associate the RS256_X509 key with the device.
6. The **Device metadata** field is optional; leave it blank.
7. Click **Create.**

You've just added a device to your registry. The RS256_X509 key appears on the **Device details** page for your device.

Run a Node.js sample to connect a virtual device and view telemetry

1. Get the Cloud IoT Core Node.js samples from GitHub. The Cloud IoT Core samples are in the `iot` directory.

```
git clone https://github.com/googleapis/nodejs-iot.git
```

2. In your cloned repository, navigate to the **`iot/mqtt_example`** directory. You'll complete the rest of these steps in this directory.

```
cd nodejs-iot/samples/mqtt_example
```

3. Copy the private key you created in the previous section (**`rsa_private.pem`**) to the current directory (**`samples/mqtt_example`**):

```
cp ../../../rsa_private.pem .
```

4. Install the Node.js dependencies:

```
npm install
```

5. Run the following command to create a subscription (https://cloud.google.com/pubsub/docs/admin#managing_subscriptions) to the registry's Pub/Sub topic, substituting your project ID:

```
gcloud pubsub subscriptions create \
    projects/PROJECT_ID /subscriptions/my-subscription \
    --topic=projects/PROJECT_ID /topics/my-device-events
```

6. Run the following command to connect a virtual device to Cloud IoT Core using the MQTT bridge, substituting your project ID. Make sure **`serverCertFile`** is set to the location where you downloaded Google's root certificate.

```
node cloudiot_mqtt_example_nodejs.js \
   mqttDeviceDemo \
   --projectId=PROJECT_ID  \
   --cloudRegion=REGION  \
   --registryId=my-registry \
   --deviceId=my-device \
   --privateKeyFile=rsa_private.pem \
   --serverCertFile=roots.pem \
   --numMessages=25 \
   --algorithm=RS256
```

The output shows that the sample device is publishing messages to the telemetry topic. Twenty-five messages are published.

★ **Note:** If you're using a Compute Engine VM or Cloud Shell, set the **`--mqttBridgePort`** flag to **`443`** when you run the device code.

7. Run the following command to read the messages published to the telemetry topic, substituting your project ID:

```
gcloud pubsub subscriptions pull --auto-ack \
    projects/PROJECT_ID/subscriptions/my-subscription
```

8. Repeat the **`subscriptions pull`** command to view additional messages.

Clean up

To avoid incurring charges to your Google Cloud account for the resources used in this quickstart, follow these steps.

1. Go to the Google Cloud Pub/Sub **Topics** page in Cloud Console.

   Go to the Google Cloud Pub/Sub Topics page (https://console.cloud.google.com/cloudpubsub/t

2. Select the checkbox next to your topic in the list, then click **Delete** at the top of the page.
3. Go to the **Registries** page in Cloud Console.

   Go to the Registries page (https://console.cloud.google.com/iot/registries)

4. Click the name of your registry in the list.
5. Click the name of your device in the list.
6. At the top of the **Device details** page, click **Delete.** Type the name of the device to confirm deletion.

   To delete a registry, you must first delete all the devices within it. If you've added any other devices to this registry, delete those too.

7. Go to the **Registries** page in Cloud Console.

Go to the **Registries** page (https://console.cloud.google.com/iot/registries)

8. Click the name of your registry in the list.
9. At the top of the **Registry details** page, click **Delete.**

What's next

- Get an overview of Cloud IoT Core (/iot/docs/concepts/overview)
- Create a device registry and devices (/iot/docs/how-tos/devices)

Last updated 2021-03-30 UTC.