

Carlo Gavazzi UWP 3.0

Getting Started Guide for AWS IoT Core

Table of Content

Document Information	1
Document revision history	1
Applicable operating system for this guide	1
Overview	2
Hardware description.....	2
User provided items	2
3 rd party purchasable items	2
Set up your development environment	2
Tools installation (IDEs, Toolchains, SDKs)	2
Set up device hardware.....	2
Set up your AWS account and permissions	2
Create resources in AWS IoT	3
Provision the UWP 3.0 with credentials	6
Verify messages in AWS IoT Core	6
Troubleshooting.....	6

Document Information

Document revision history

Revision	Date	Description
REV01	2023/08/22	Publish document

Applicable operating system for this guide

This guide is applicable to all the operating system which supports a browser.

Overview

This document describes how to connect a Carlo Gavazzi UWP 3.0 gateway to AWS IoT-Core.

Note: for further details about UWP 3.0 go to our website clicking [here](#)

Hardware description

The following UWP 3.0 models have been tested

Device name	Datasheet link	Installation manual link	Website page
UWP30RSEXXX	UWP 3.0 DS	UWP 3.0 IM	Link
UWP30RSEXXXSE		UWP 3.0 SE IM	Link

User provided items

None

3rd party purchasable items

None

Set up your development environment

Tools installation (IDEs, Toolchains, SDKs)

None

Set up device hardware

Following are the main steps:

1. Perform the UWP 3.0 commissioning
Note: for further details, read the [installation manual](#)
2. Connect the UWP 3.0 to the Internet.
Notes:
 - For further details, read the [quick guide connection](#)
 - Remember to set the DNS server properly from the **System settings** menu of the controller web app

Set up your AWS account and permissions

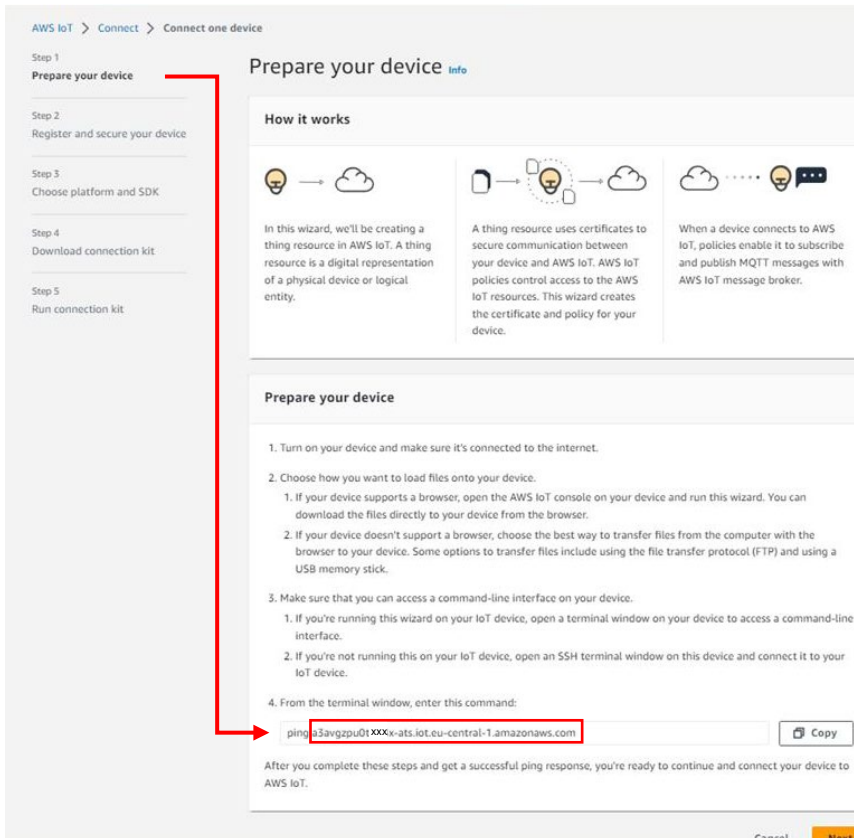
Following are the main steps:

3. Log into your AWS account
Note: click [here](#) for more information about how to set up an AWS IoT-Core account
4. Create AWS IoT resources for your device.
Notes:
A 5-step procedure will start (AWS IoT > Connect > Connect one device)
To get started, follow the steps outlined in the sections below:
 - [Sign up for an AWS account](#)
 - [Create an administrative user](#)
 - [Open the AWS IoT console](#)*Pay special attention to the Notes.*
Click [here](#) for more information about how to create AWS IoT resources.

Create resources in AWS IoT

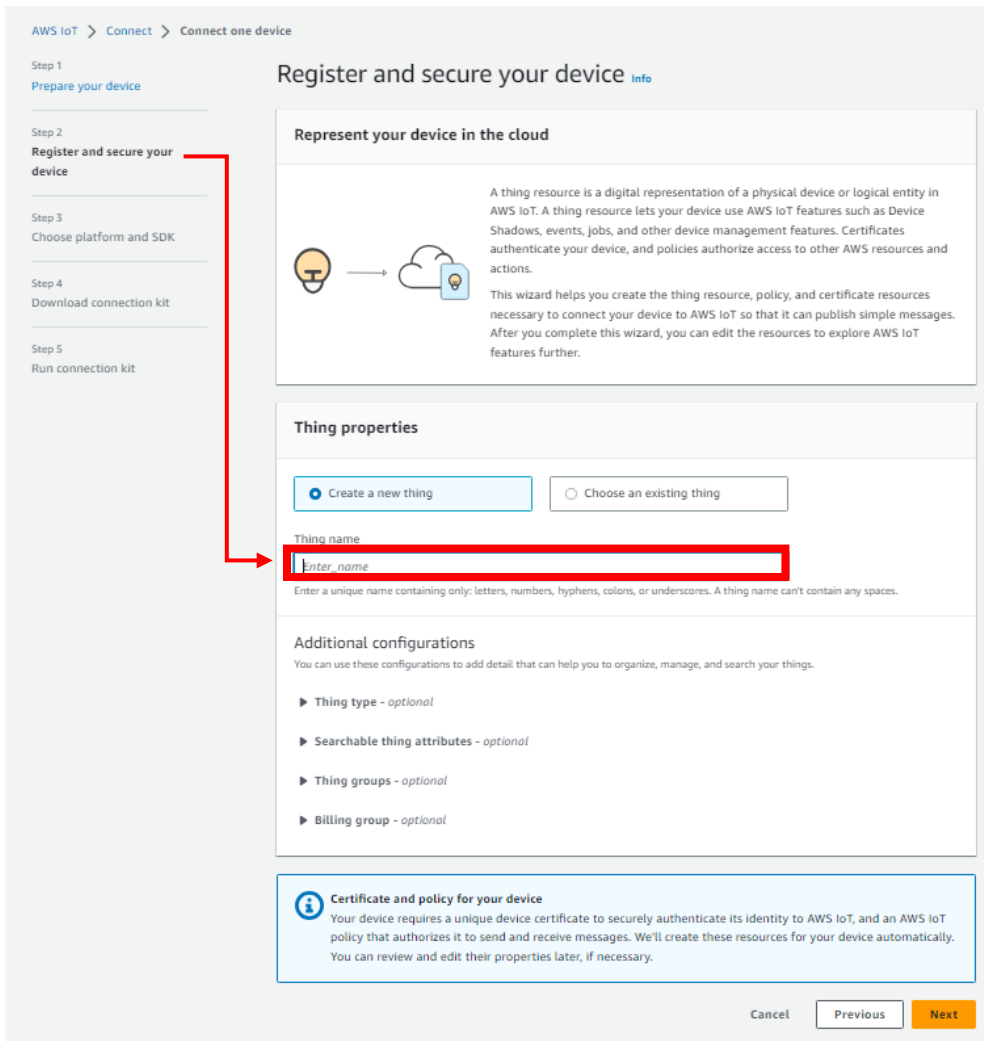
Following are the main steps:

- In **Step 1 – Prepare your device**, from point 4 copy the string, and paste it into a separate sheet deleting the word “ping” as shown below.
This is the **connection string** that you will use in step 16



The screenshot shows the AWS IoT console interface for the 'Prepare your device' wizard. On the left, a sidebar lists five steps: Step 1 (Prepare your device), Step 2 (Register and secure your device), Step 3 (Choose platform and SDK), Step 4 (Download connection kit), and Step 5 (Run connection kit). The main content area is titled 'Prepare your device' and includes an 'Info' link. Under 'How it works', there are three explanatory panels with icons. Below this, a list of instructions guides the user through connecting the device. Step 4 instructs the user to enter a terminal command in a text box. The command is 'ping a3avgzpu0txxx-ats.iot.eu-central-1.amazonaws.com', where 'a3avgzpu0txxx' is highlighted with a red box. A 'Copy' button is next to the text box. At the bottom right, there are 'Cancel' and 'Next' buttons.

- In **Step 2 - Register and secure your device**, enter a **thing name** and copy it into a separate sheet.
This is the **client ID** and **the Topic** that you will use in step 16



7. In **Step 3 - Choose platform and SDK**, set:
 - **Linux** as device platform operating system
 - **Node.js** as AWS IoT Device SDK

For UWP 3.0 is not necessary to run the connection kit (skip step 5 of the **AWS IoT > Connect > Connect one device** procedure)

8. To provision the UWP 3.0 with a signed AWS certificate, from your AWS IoT-Core account go to **Security > Certificates**
9. Click **Add certificate > Create certificate**
10. From **Certificate**, select **Auto-generate new certificate**, set the certificate status to **Active** and click **create** (as shown below)

AWS IoT > Security > Certificates > Create certificate

Create certificate [Info](#)

Certificates authenticate devices and clients so that they can connect to AWS IoT. Your device won't be able to connect to AWS IoT without authentication and an appropriate policy.

Certificate

Auto-generate new certificate (recommended)
Generate a new certificate, public key, and private key using AWS IoT's certificate authority and register it with AWS IoT.

Create certificate with certificate signing request (CSR)
Upload your own certificate signing request (CSR) file to create and register a certificate that's based on a private key you own.

Certificate status

Assign the initial state of the new certificate. The certificate must be active before it can be used to connect to AWS IoT. You can change its status later in the certificate's detail page.

Inactive
A device won't be able to connect to AWS using this certificate until it's activated.

Active
A device will be able to connect to AWS using this certificate immediately after you create it.

Cancel Create

11. Download the **Device certificate** and **Private key file** as shown below.
Note: you will use these files in step 16.

Download certificates and keys

Download certificates and keys

Download and install the certificate and key files to your device so that it can connect securely to AWS IoT. You can download the certificate now, or later, but the key files can only be downloaded now.

Device certificate
1e12bdf83dd...te.pem.crt Download

Key files

The key files are unique to this certificate and can't be downloaded after you leave this page. Download them now and save them in a secure place.

⚠ This is the only time you can download the key files for this certificate.

Public key file
1e12bdf83dded8dc3730a4c...1c70e67-public.pem.key Download

Private key file
1e12bdf83dded8dc3730a4c...c70e67-private.pem.key Download

Root CA certificates

Download the root CA certificate file that corresponds to the type of data endpoint and cipher suite you're using. You can also download the root CA certificates later.

Amazon trust services endpoint
RSA 2048 bit key: Amazon Root CA 1 Download

Amazon trust services endpoint
ECC 256 bit key: Amazon Root CA 3 Download

If you don't see the root CA certificate that you need here, AWS IoT supports additional root CA certificates. These root CA certificates and others are available from our developer guides.

Continue

Provision the UWP 3.0 with credentials

Following are the main steps:

12. Log into the UWP 3.0 Web App
13. From the **Navigation bar**, click ☰ to open the Main menu.
14. From the **Services** menu, select the **AWS IoT service** to open the configuration page.
15. From the **Service configuration** tile, click ▼ (under **Service**) to select **Enable**.


16. In the same tile, add the:
 - **Connection string** (saved in step 5)
 - **Client ID** (saved in step 6)
 - **Topic** (saved in step 6)
 - **Security certificates**

*Note: click **Upload certificate files** to upload the **Device certificate** and the **private key file** saved in step 12*

- **Upload interval.**

Note: The Start date is not available when the service is enabled.

17. Click **Select variables** to choose the devices that the Data Push service has to consider
*Note: this menu shows the devices that have been enabled to log data in the UWP 3.0 database.
 For more information about how to configure the database, read the [UWP 3.0 Tool manual](#)*

18. Click  to save the configuration.

19. From the **Information** tile, check on the service status.
*The green **Status** icon informs you that the procedure has been completed successfully.
 Click **Show logs – OK** to open the list of successfully loaded data.*

Verify messages in AWS IoT Core

The AWS IoT MQTT client allows to view the data sent by your device.
 Click [here](#) for more information about the client.

Troubleshooting

- From UWP 3.0 Web App, from the **Information** tile in **Services > AWS IoT service**, check on the service status.

Element	Description
Status	Service status: ● Active / ○ Inactive / ● Disconnected
Last data transmission	Date/time of the last data transmission
Show logs - OK	Logs lost successfully loaded
Show logs - Errors	Logs list errors

- For more information about AWS Troubleshooting click [here](#)