



QUICK START GUIDE

**VIA SOM-5000**

AWS IoT Core



## Copyright

Copyright © 2024 VIA Technologies Incorporated. All rights reserved.

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise without the prior written permission of VIA Technologies, Incorporated.

## Trademarks

All brands, product names, company names, trademarks and service marks are the property of their respective holders.

## Disclaimer

VIA Technologies makes no warranties, implied or otherwise, in regard to this document and to the products described in this document. The information provided in this document is believed to be accurate and reliable as of the publication date of this document. However, VIA Technologies assumes no responsibility for the use or misuse of the information (including use or connection of extra device/equipment/add-on card) in this document and for any patent infringements that may arise from the use of this document. The information and product specifications within this document are subject to change at any time, without notice and without obligation to notify any person of such change.

VIA Technologies, Inc. reserves the right to make changes to the products described in this manual at any time without prior notice.

## Revision History

Version	Date	Remarks
1.00	02/09/2024	Initial release

## Table of Contents

1.	Introduction.....	1
1.1	VIA SOM-5000 Device.....	1
1.2	Directing Data from a VIA SOM-5000 Device to AWS IoT Core.....	2
2.	Connecting to AWS IoT Core.....	3
2.1	Introduction .....	3
2.2	Set up your AWS Account and Permissions .....	4
2.3	Create Resources in AWS IoT .....	4
2.4	Image Installation.....	11
2.5	Connecting the VIA SOM-5000 Device to AWS IoT Core.....	12
2.5.1	Connecting the Debug Console.....	12
2.5.2	Connecting with the Android 13.0 EVK.....	13
2.6	Debugging .....	16
2.7	Troubleshooting .....	16

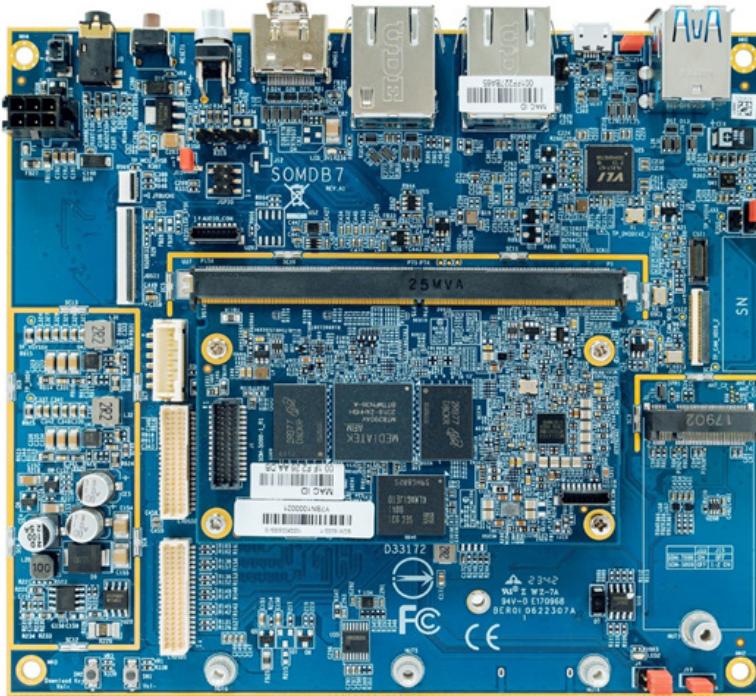
# 1. Introduction

This document provides instructions on how to set up AWS IoT Core to connect with a VIA SOM-5000 device.

## 1.1 VIA SOM-5000 Device

Powered by the high performance and energy-efficient octa-core MediaTek Genio 700 SoC, the VIA SOM-5000 Edge AI board delivers blistering edge processing and multimedia performance, while combining advanced AI technology for display, object recognition, and voice, with rich wireless and I/O connectivity features. Besides dual-core ARM Cortex-A78 and hexa-core ARM Cortex-A55 processors, the MediaTek Genio 700 platform also features an integrated AI processor providing up to 4.0 TOPS for deep learning, neural network acceleration, and computer vision applications, including facial recognition, object identification, and OCR.

Based on the 3.5" SBC 15.1cm x 13.4cm form factor, the VIA SOM-5000 board provides a wealth of camera and display integration options, including support for two MIPI CSI-2 cameras, a MIPI DSI display, plus, two HDMI and LVDS displays each. High-speed connectivity can be enabled through two Gigabit Ethernet ports, an integrated SIM card slot, and M.2 slots for optional dual-band Wi-Fi 6 and 4G LTE. Two USB 3.1 ports, an ADC/GPIO pin header, and a MicroSD card slot allow additional I/O configuration.



Refer to the datasheet and user manual on the [SOM-5000 product page](#) for detailed information.

## 1.2 Directing Data from a VIA SOM-5000 Device to AWS IoT Core

To direct data from VIA SOM-5000 devices to a cloud implementation, the AWS IoT Core Web Service must be set up and configured to receive data coming from VIA SOM-5000 devices.

Follow the steps below to set up your AWS services in order to connect a VIA SOM-5000 device to an AWS backend:

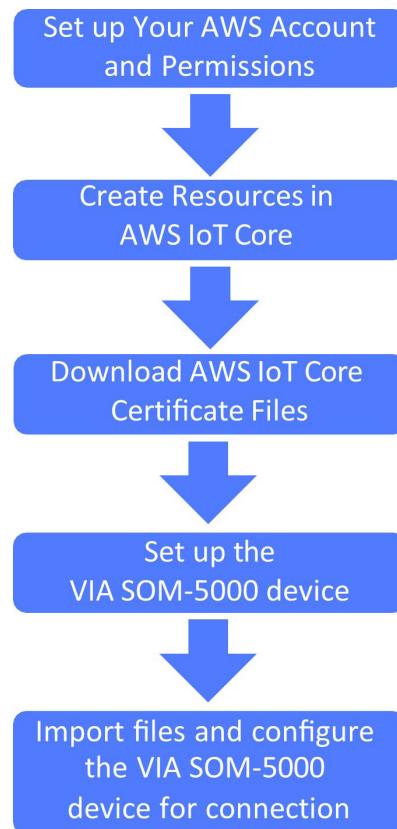
- Follow the steps listed in [Section 2.3](#) to register an AWS IoT Thing and to acquire the AWS IoT certificate files for the VIA SOM-5000 devices.
  - Acquire the certificate (\*-certificate.pem) and the private key (\*-private.pem.key) files associated with the IoT thing created, as described in [Step 7 of Section 2.3](#).
  - Acquire the Device Data Endpoint described in [Step 9 of Section 2.3](#).

## 2. Connecting to AWS IoT Core

### 2.1 Introduction

This section provides instructions on how to establish a connection between VIA SOM-5000 devices and AWS IoT Core, including how to add and manage VIA SOM-5000 devices as an IoT Core device on AWS, set up the device, and configure the VIA IoT application.

To register an AWS IoT thing and acquire the AWS IoT certificate files for the VIA SOM-5000 device, follow the steps below:



## 2.2 Set up your AWS Account and Permissions

Refer to instructions in the following sections on webpage <https://docs.aws.amazon.com/iot/latest/developerguide/setting-up.html> to set up an AWS Account:

- [Sign up for an AWS account](#)
- [Create a user and grant permissions](#)
- [Open the AWS IoT console](#)



Sign in as IAM user

Account ID (12 digits) or account alias

IAM user name

Password

Remember this account

**Sign in**

Sign in using root user email

Forgot password?



Pay special attention to the Notes on the AWS account setup webpage.

## 2.3 Create Resources in AWS IoT

Refer to instructions in the following sections on webpage <https://docs.aws.amazon.com/iot/latest/developerguide/create-iot-resources.html> to provision resources for your VIA SOM-5000 device:

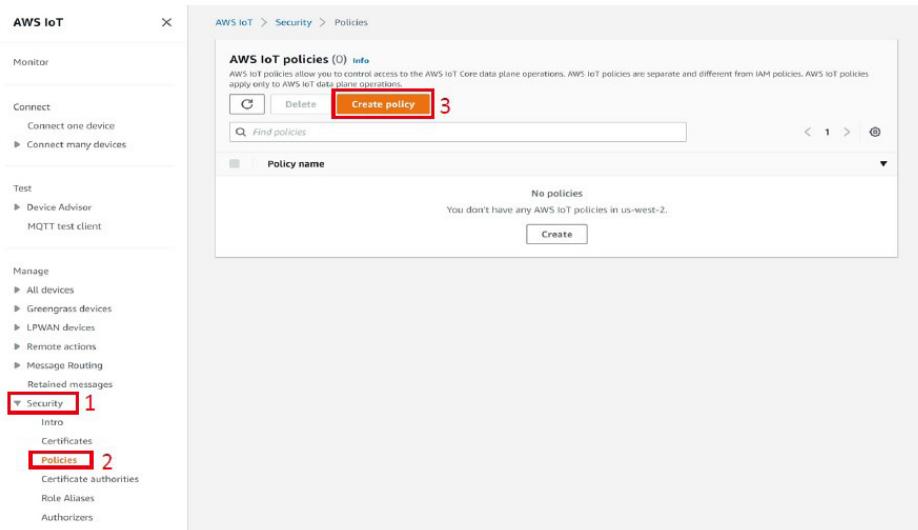
- [Create an AWS IoT policy](#)
- [Create a thing object](#)

Pay special attention to the Notes on the AWS IoT resource creation webpage.

The following instructions demonstrate how to create an AWS IoT Policy and a thing object, and acquiring the required AWS IoT certificate files and Device Data Endpoint for the VIA SOM-5000 device.

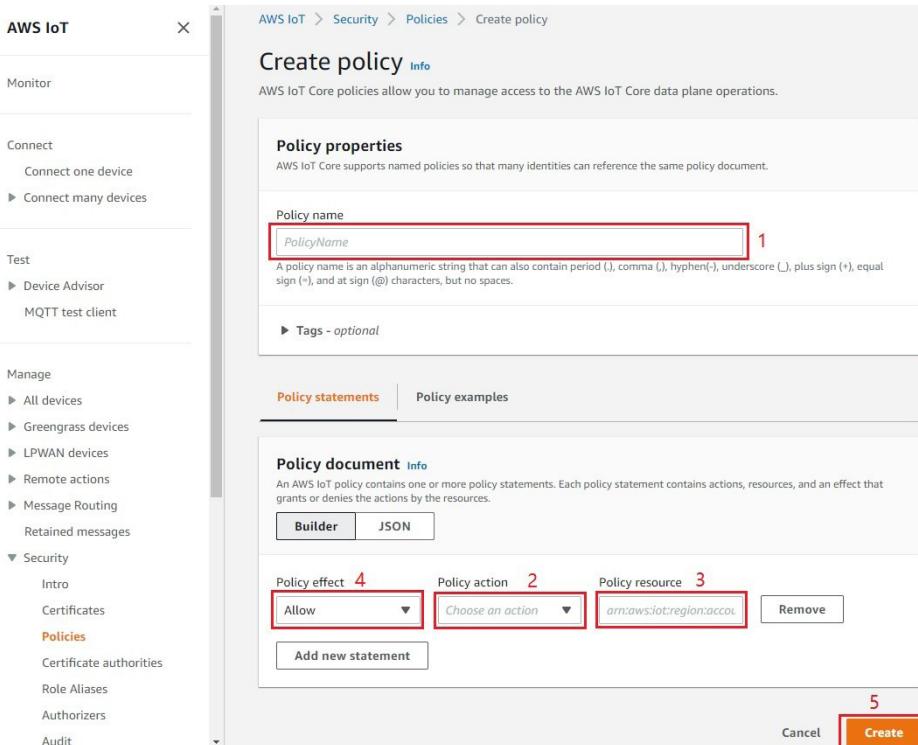
## Step 1

Click on 'Security/Policies' in the left panel and then click on the 'Create policy' button in the right panel.

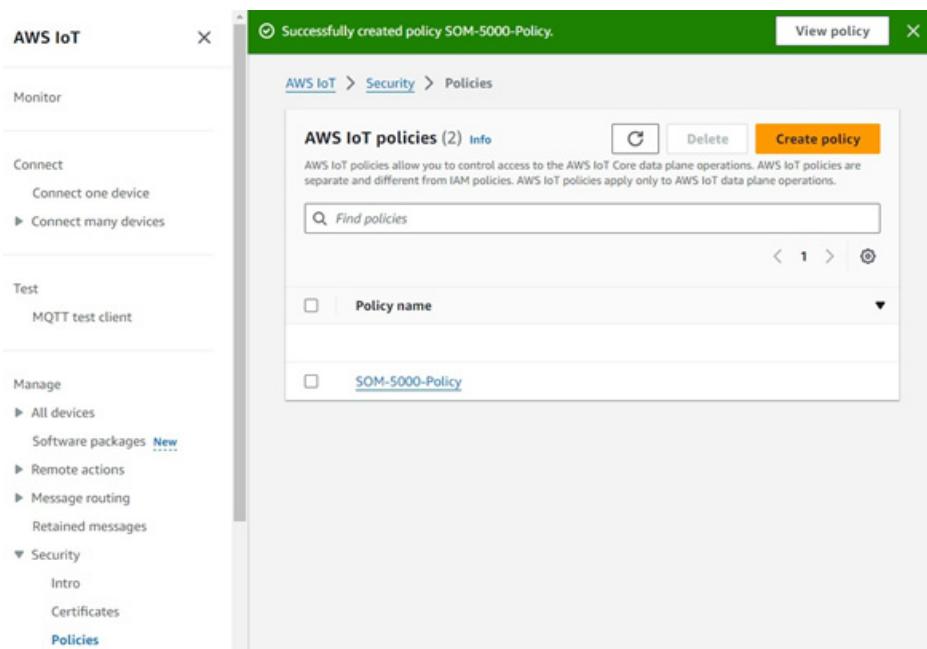


## Step 2

In the "Create policy" form, fill in the "Policy name" in the "Policy properties" section. In the "Policy document" section, select the "Policy effect" and "Policy action", and fill in the "Policy resource" name. Then, click the 'Create' button to submit.



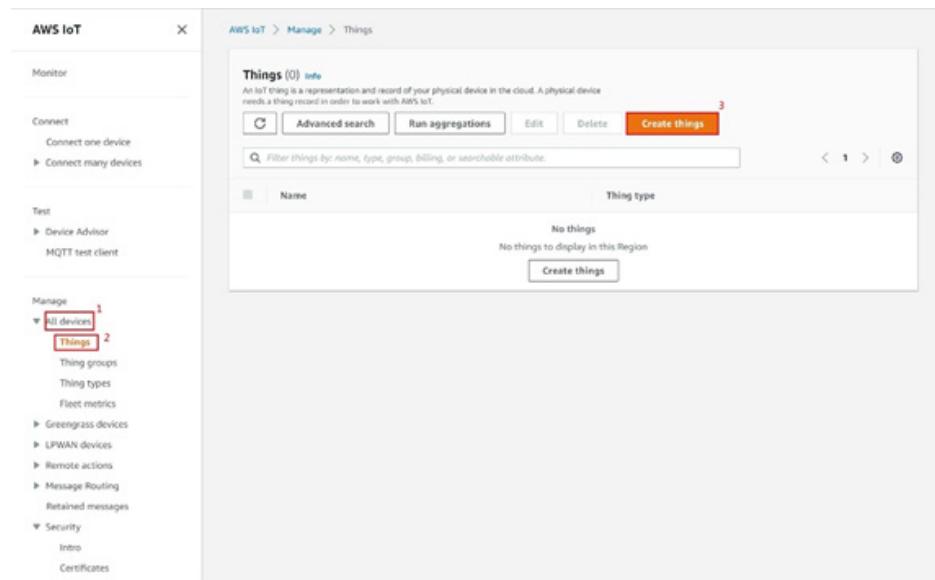
After clicking the 'Create' button, the new policy's name will be listed on the right panel.



The screenshot shows the AWS IoT Policies page. The left sidebar has a 'Policies' section selected. The main content area shows a success message: 'Successfully created policy SOM-5000-Policy.' Below this, it says 'AWS IoT policies (2) Info'. A table lists two policies: 'Policy name' and 'SOM-5000-Policy'.

### Step 3

To create a new 'IoT Thing' resource with the new policy and to create certificate files for this resource, click on 'All devices/Things' in the left panel and then click on the 'Create things' button in the right panel.



The screenshot shows the AWS IoT Things page. The left sidebar has a 'Things' section selected. The main content area shows a table with columns 'Name' and 'Thing type'. A red box highlights the 'Create things' button at the bottom of the table.

Next, select 'Create single thing' and click on "Next".

AWS IoT > Manage > Things > Create things

## Create things Info

A thing resource is a digital representation of a physical device or logical entity in AWS IoT. Your device or entity needs a thing resource in the registry to use AWS IoT features such as Device Shadows, events, jobs, and device management features.

**Number of things to create**

**Create single thing**  
Create a thing resource to register a device. Provision the certificate and policy necessary to allow the device to connect to AWS IoT.

**Create many things**  
Create a task that creates multiple thing resources to register devices and provision the resources those devices require to connect to AWS IoT.

2

Cancel **Next**

#### Step 4

Under 'Thing properties' in the right panel, enter the 'Thing Name'. Under 'Device Shadow', select 'Unnamed shadow (classic)' and click on the 'Next' button to continue.

AWS IoT > Manage > Things > Create things > Create single thing

**Step 1**  
**Specify thing properties**

Step 2 - optional  
Configure device certificate

Step 3 - optional  
Attach policies to certificate

### Specify thing properties Info

A thing resource is a digital representation of a physical device or logical entity in AWS IoT. Your device or entity needs a thing resource in the registry to use AWS IoT features such as Device Shadows, events, jobs, and device management features.

**Thing properties Info**

Thing name  
**SOM-5000** 1

Enter a unique name containing only: letters, numbers, hyphens, colons, or underscores. A thing name can't contain any spaces.

**Additional configurations**

You can use these configurations to add detail that can help you to organize, manage, and search your things.

- ▶ Thing type - *optional*
- ▶ Searchable thing attributes - *optional*
- ▶ Thing groups - *optional*
- ▶ Billing group - *optional*
- ▶ Packages and versions - *optional*

**Device Shadow Info**

Device Shadows allow connected devices to sync states with AWS. You can also get, update, or delete the state information of this thing's shadow using either HTTPS or MQTT topics.

No shadow

Named shadow  
Create multiple shadows with different names to manage access to properties, and logically group your devices properties.

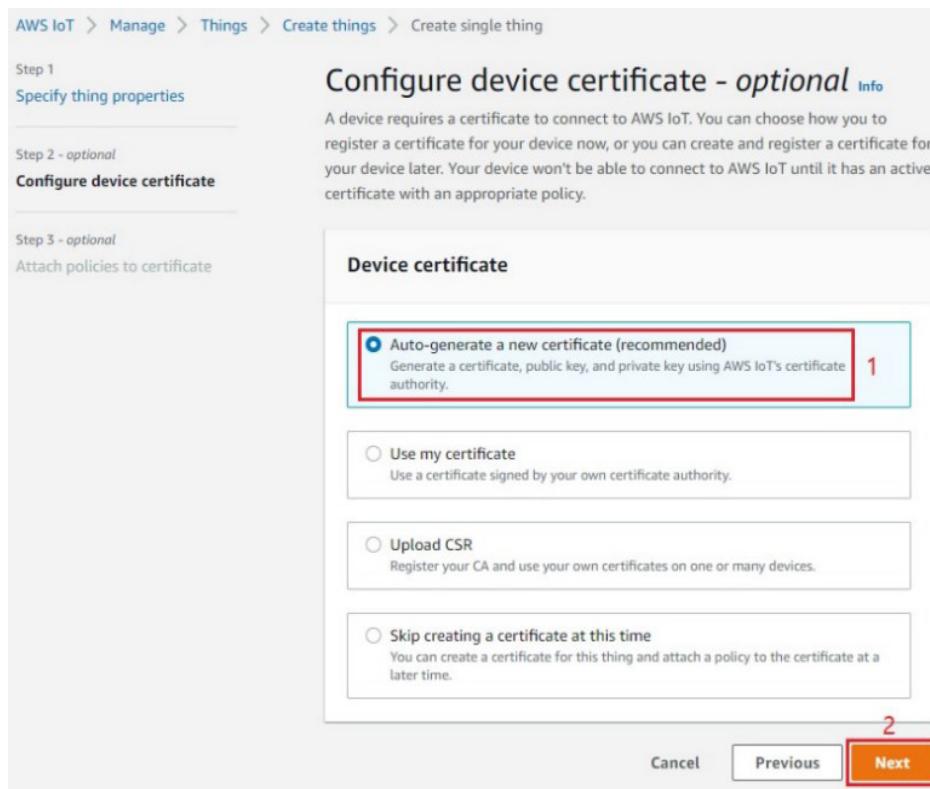
**Unnamed shadow (classic)**  
A thing can have only one unnamed shadow. 2

▶ Edit shadow statement - *optional*

Cancel 3 **Next**

## Step 5

Under 'Device certificate', select 'Auto-generate a new certificate (recommended)' and click on the 'Next' button to configure a device certificate for your VIA SOM-5000 device.



AWS IoT > Manage > Things > Create things > Create single thing

Step 1  
Specify thing properties

Step 2 - optional  
Configure device certificate

Step 3 - optional  
Attach policies to certificate

**Configure device certificate - optional** Info

A device requires a certificate to connect to AWS IoT. You can choose how you to register a certificate for your device now, or you can create and register a certificate for your device later. Your device won't be able to connect to AWS IoT until it has an active certificate with an appropriate policy.

**Device certificate**

**Auto-generate a new certificate (recommended)**  
Generate a certificate, public key, and private key using AWS IoT's certificate authority. 1

**Use my certificate**  
Use a certificate signed by your own certificate authority.

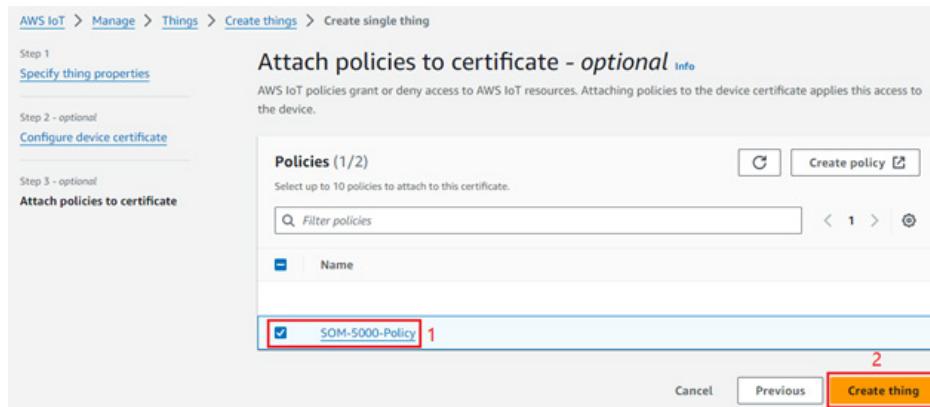
**Upload CSR**  
Register your CA and use your own certificates on one or many devices.

**Skip creating a certificate at this time**  
You can create a certificate for this thing and attach a policy to the certificate at a later time.

Cancel Previous Next 2

## Step 6

Click on the new policy's name and click on the 'Create Thing' button to attach policies to the device certificate for your VIA VAB-3000 device and to complete creating a new 'IoT Thing'.



AWS IoT > Manage > Things > Create things > Create single thing

Step 1  
Specify thing properties

Step 2 - optional  
Configure device certificate

Step 3 - optional  
Attach policies to certificate

**Attach policies to certificate - optional** Info

AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the device certificate applies this access to the device.

**Policies (1/2)**

Select up to 10 policies to attach to this certificate.

**SOM-5000-Policy** 1

Cancel Previous Create thing 2

## Step 7

Download the device certificate (CertFile), the private key file (PvkFile) and the 2048-bit Amazon Root CA 1 certificate (CAFile) to a local folder and click on 'Done' to complete the certificate creation process for your VIA SOM-5000 device.

### 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

...te.pem.crt

 Download

1

#### 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

...-public.pem.key

 Download

2

Private key file

...-private.pem.key

 Download

3

#### 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. [Learn about using CA certificates for authentication](#)

Amazon trust services endpoint

RSA 2048 bit key: Amazon Root CA 1

 Download

4

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

## Step 8

Click on the created IoT thing's name under 'Things' to view your VIA SOM-5000 device's Thing and Device Shadow details, including the MQTT topic prefix.



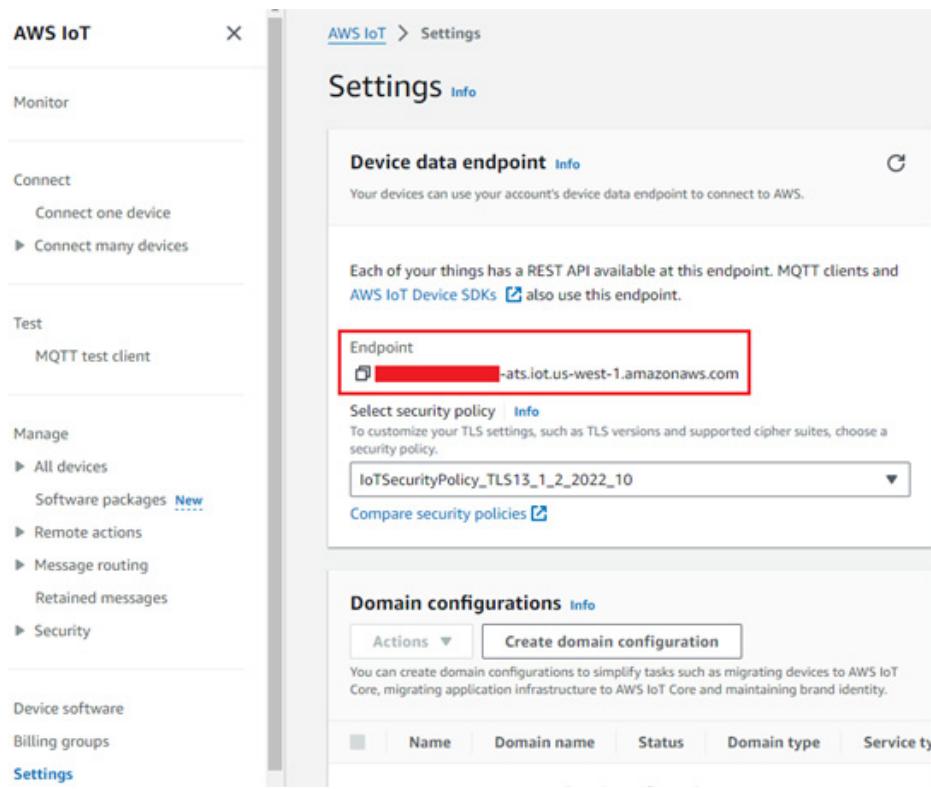
### Note:

The MQTT topic prefix is for binding your VIA SOM-5000 device with AWS IoT Core Web services. The VIA SOM-5000 device will use the specified MQTT topic to interact with AWS IoT Core Web services. Some [MQTT topics listed on the AWS webpage](#) are reserved.

The screenshot shows the AWS IoT Core console interface. On the left, the navigation menu includes 'Monitor', 'Connect' (with 'Connect one device' and 'Connect many devices' options), 'Test' (with 'MQTT test client'), and 'Manage' (with 'All devices', 'Thing groups', and 'Thing types' options). The 'Things' option under 'Manage' is selected, showing a list of things with a count of 2. The first item in the list is 'SOM-5000', which is highlighted. The main content area shows a 'Things (2) Info' section with a 'Create things' button. Below this is a table with columns for 'Name' and 'Thing type', showing 'SOM-5000' and '-' respectively. The 'Thing details' section for 'SOM-5000' shows the 'Name' as 'SOM-5000', 'ARN' as 'arn:aws:iot:us-west-1:367273022368:thing/SOM-5000', and 'Type' as '-'. The 'Device Shadows' section shows a table with a single entry for 'Classic Shadow' with the MQTT topic prefix '\$aws/things/SOM-5000/shadow'.

### Step 9

In the left "AWS-IoT" panel, select "Settings". The "Endpoint" is displayed in the "Device data endpoint" section.



The screenshot shows the AWS IoT Settings page. On the left, a sidebar lists various options: Monitor, Connect (with 'Connect one device' and 'Connect many devices' sub-options), Test (with 'MQTT test client'), Manage (with 'All devices', 'Software packages', 'Remote actions', 'Message routing', 'Retained messages', and 'Security' sub-options), Device software, Billing groups, and Settings (which is selected and highlighted in blue). The main content area is titled 'Settings' and contains a section for 'Device data endpoint'. It states: 'Your devices can use your account's device data endpoint to connect to AWS.' Below this, it says: 'Each of your things has a REST API available at this endpoint. MQTT clients and AWS IoT Device SDKs also use this endpoint.' A red box highlights the 'Endpoint' field, which contains the URL 'https://[REDACTED]-ats.iot.us-west-1.amazonaws.com'. Below the endpoint, there is a 'Select security policy' dropdown set to 'IoTSecurityPolicy\_TLS13\_1\_2\_2022\_10' and a 'Compare security policies' link. Another section titled 'Domain configurations' is partially visible at the bottom.

## 2.4 Image Installation

The VIA SOM-5000 BSP supports Android 13.0 operating systems. To enable hardware functionality, download the VIA SOM-5000 Android 13.0 EVK package, and refer to the corresponding EVK Quick Start Guide included in the package to flash the device.



The screenshot shows the VIA SOM-5000 BSP software download page. It features a large image of a person's hand interacting with a computer monitor displaying code. The page is titled 'VIA SOM-5000 BSP' and describes the BSP as supporting Android 13, Yocto 4.0, and Debian 12, with software customization services available. It lists two download options: 'Android 13.0 EVK' (selected, highlighted in red) and 'Yocto 4.0 EVK [Coming Soon]'. Below the download links, a note states: 'Important: By downloading these software packages, you indicate your acceptance of the VIA Software License agreement. If you do not agree with any of the terms and conditions, do not continue to download the software.' The download link for the Android 13.0 EVK is: <https://www.via.com/Software/VIA-SOM-5000-Android-13.0-EVK>.

## 2.5 Connecting the VIA SOM-5000 Device to AWS IoT Core

### 2.5.1 Connecting the Debug Console

The first step is to connect the VIA SOM-5000 device to the debug console for setting up a connection between the VIA SOM-5000 device and AWS IoT Core. The debug cable of the VIA SOM-5000 and a serial port communication program such as PuTTY or Tera Term are required for the setup.

Follow the steps below to connect the VIA SOM-5000 device to the debug console:

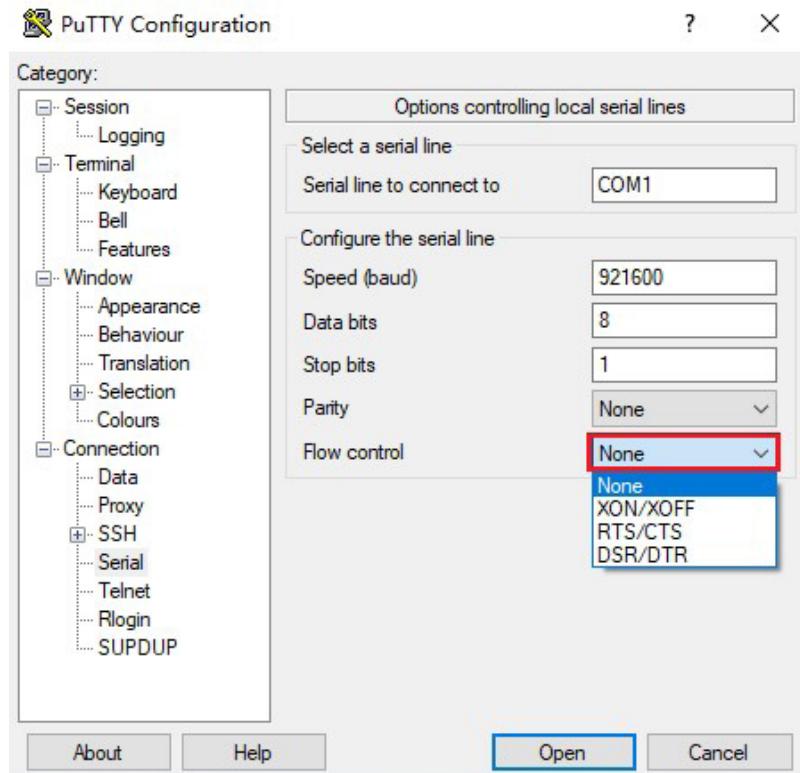
#### Step 1

Refer to section 3.1 of the VIA SOM-5000 Android 13.0 EVK Quick Start Guide for instructions on connecting Windows host machine and the VIA SOM-5000 device through the provided debug cable.



#### Step 2

Use a serial port communication program such as PuTTY or Tera Term to connect the debug console. Set the console Baud Rate to "921600".



#### Step 3

Power on the VIA SOM-5000 device to initiate the boot process.

**Step 4**

When the boot process has completed, log in to the debug console.

**Step 5**

Refer to the section 2.5.2 to connect the VIA SOM-5000 device to your AWS IoT Core Web service.

## 2.5.2 Connecting with the Android 13.0 EVK

The VIA SOM-5000 Android 13.0 EVK includes an "IoTDeviceSDKSample" application which establishes a connection between the VIA SOM-5000 device and the AWS IoT Core Web service.

This section guides developers on how to enable and run the "IoTDeviceSDKSample" application.

**Step 1**

Edit the "iotcoreConfig.json" settings file and modify the following values based on the AWS IoT Core setup:

- **thingName:** Insert the defined thing's name.
- **ca:** Input the content text of the thing's Amazon Root CA certificate.
- **cert:** Input the content text of the thing's Certificate file.
- **key:** Input the content text of the thing's Private Key file.
- **ReqRetry:** Number of retries for establishing the connection between the device and the IoT Core server when the connection is disrupted or not successful.
- **topic:** The MQTT topic.
- **endpoint:** Input the domain name from the Device Data Endpoint obtained in [Step 9 of Section 2.3](#).

Save the settings file to a USB flash drive and plug it to the device's USB port.

```
{  
    "thingName" : "SOM-5000",  
    "ca" : "-----BEGIN CERTIFICATE-----MIIDQTCCAimgAwIBAgITBmyfz5m/jAo54  
    "cert" : "-----BEGIN CERTIFICATE-----MIIDWTCCAkGgAwIBAgIUGUdv5laxeUEzg  
    "key" : "-----BEGIN RSA PRIVATE KEY-----MIIEpAIBAAKCAQEApQk3BV5gtgIeT  
    "ReqRetry" : "10",  
    "topic" : "$aws/things/SOM-5000/shadow/update",  
    "endpoint": "ats.iot.ap-northeast-1.amazonaws.com"  
}
```

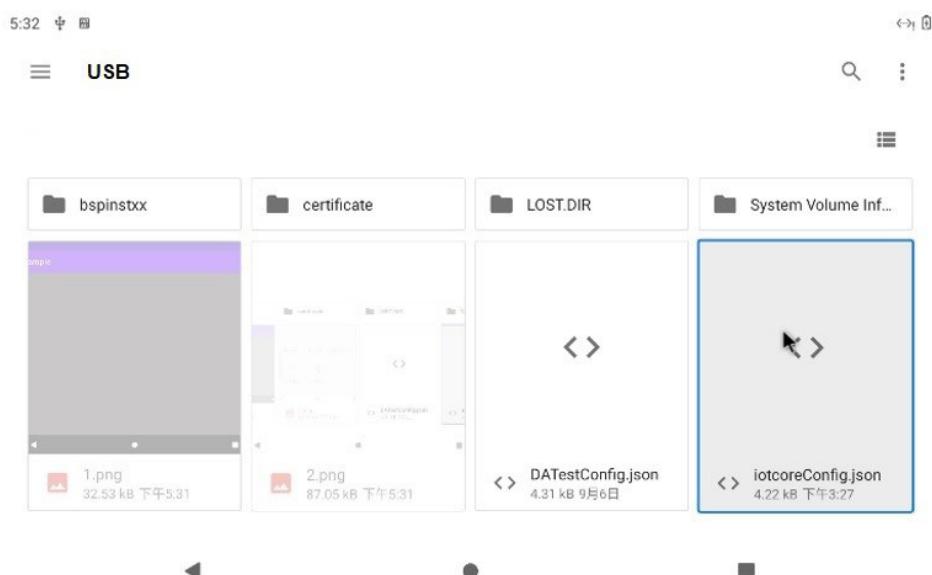
**Step 2**

Run the "IoTDeviceSDKSample" application as shown below.

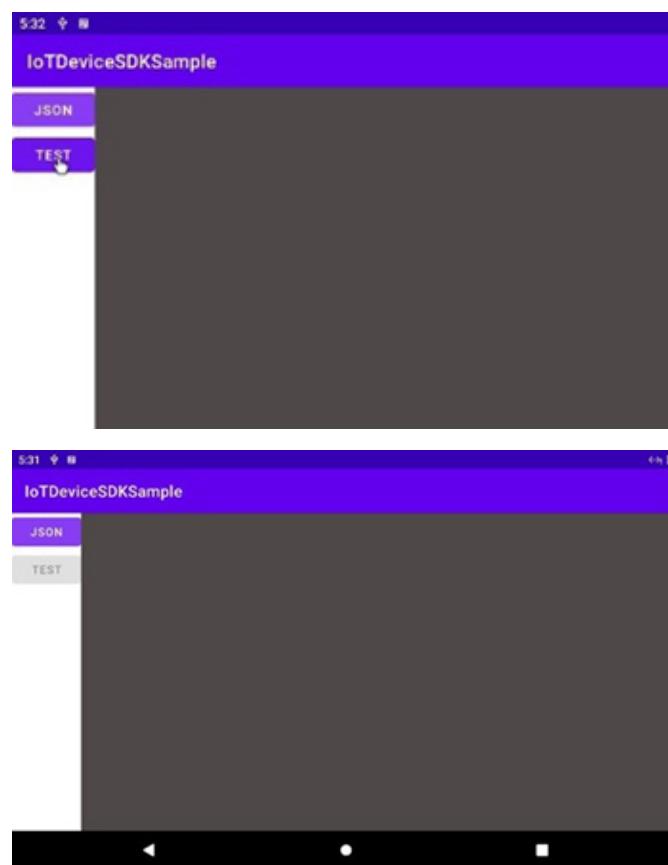


### Step 3

Click the "JSON" button to select the "iotcoreConfig.json" created in Step 1 from USB flash drive.



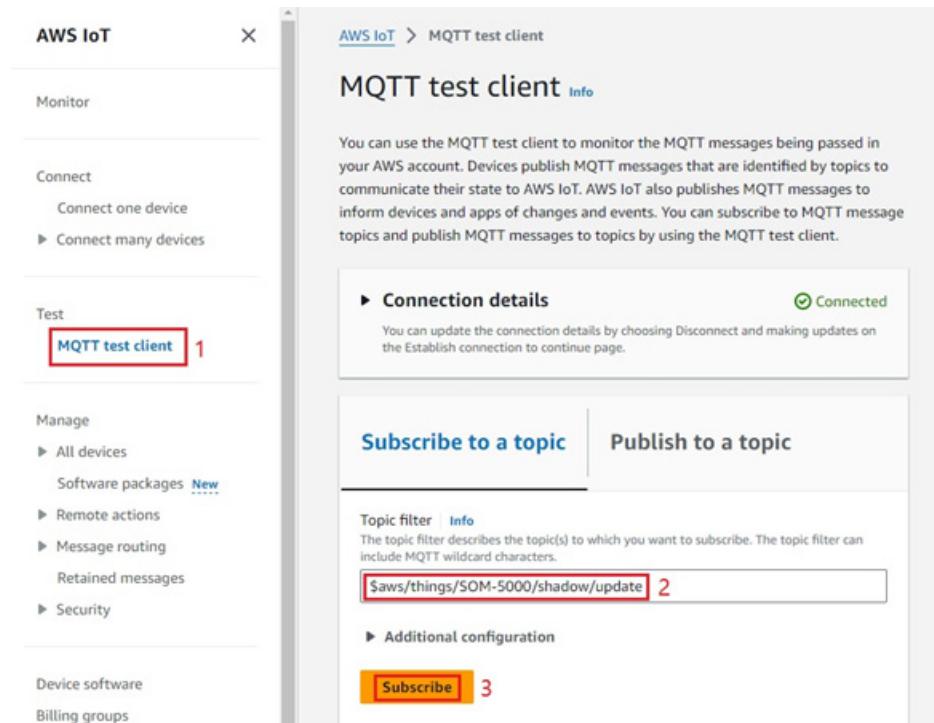
Click the "TEST" button to run the MQTT subscription publishing process as shown below.



#### Step 4

Verify the "IoTSDKSample" application using the AWS IoT Core Web service console. Click on the 'MQTT test client' tab.

To subscribe to the interaction topic with the MQTT topic prefix obtained in [step 8 of Section 2.3](#) "\$aws/things/ThingName/shadow/update", enter a topic filter name and click the 'Subscribe' button in the 'Subscribe to a topic' tab in the right panel. The value of "thingName" must be the same as the value of "iotcoreConfig.json" settings file.



AWS IoT

MQTT test client

MQTT test client

Connection details

Connected

Subscribe to a topic

Publish to a topic

Topic filter

\$aws/things/SOM-5000/shadow/update

Subscribe

The last updated message will be shown in the AWS IoT MQTT test client.



订阅

\$aws/things/SOM-5000/shadow/update

消息负载

{  
  "message": "Hello from AWS IoT console"  
}

其他配置

发布

\$aws/things/SOM-5000/shadow/update

Hello from IOT Device.2024-07-22T03:32:47.520331

属性

## 2.6 Debugging

Open a console (e.g. Putty) and configure as described in Section 2.5.1. The boot up messages will be presented with a command line interface as well as debug output. Use busybox commands "in /bin" to set up and debug the VIA SOM-5000 device.

## 2.7 Troubleshooting

Check the table below for troubleshooting common AWS IoT Core connection issues that may arise during development:

Issue	Solution
Device does not connect to the Internet	<p>Confirm that the SIM card is inserted properly.</p> <p>Confirm that the SIM card is activated by the service provider.</p> <p>Verify the APN settings.</p>
Device does not connect to AWS	<p>Confirm that the date and time is synchronized.</p> <p>Verify the AWS endpoint in the "iot- coreConfig.json" (Android 13.0) settings file.</p> <p>Verify that appropriate certificates and keys are loaded on the VIA SOM-5000 device.</p> <p>Verify the IoT Core thing and policies set on AWS IoT Core.</p>







---

 **Taiwan Headquarters**

1F, 531 Zhong-zheng Road,  
Xindian Dist., New Taipei City 231  
Taiwan

Tel: 886-2-2218-5452  
Fax: 886-2-2218-9860  
Email: [embedded@via.com.tw](mailto:embedded@via.com.tw)

 **USA**

940 Mission Court  
Fremont, CA 94539,  
USA  
  
Tel: 1-510-687-4688  
Fax: 1-510-687-4654  
Email: [embedded@viatech.com](mailto:embedded@viatech.com)

 **Japan**

3-15-7 Ebisu MT Bldg. 6F,  
Higashi, Shibuya-ku  
Tokyo 150-0011  
Japan  
  
Tel: 81-3-5466-1637  
Fax: 81-3-5466-1638  
Email: [embedded@viatech.co.jp](mailto:embedded@viatech.co.jp)

 **China**

Tsinghua Science Park Bldg. 7  
No. 1 Zongguancun East Road,  
Haidian Dist., Beijing, 100084  
China  
  
Tel: 86-10-59852288  
Fax: 86-10-59852299  
Email: [embedded@viatech.com.cn](mailto:embedded@viatech.com.cn)

 **Europe**

Email: [embedded@via-tech.eu](mailto:embedded@via-tech.eu)