AmebaPro2 Amazon FreeRTOS-LTS - Getting Started Guide v1.3.2

Realtek Semiconductor Corp.
No. 2, Innovation Road II, Hsinchu Science Park, Hsinchu 300, Taiwan
Tel.: +886-3-578-0211. Fax: +886-3-577-6047
www.realtek.com

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

COPYRIGHT

©2022 Realtek Semiconductor Corp. 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 without the written permission of Realtek Semiconductor Corp.

DISCLAIMER

Please Read Carefully:

Realtek Semiconductor Corp., (Realtek) reserves the right to make corrections, enhancements, improvements and other changes to its products and services. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

Reproduction of significant portions in Realtek data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Realtek is not responsible or liable for such reproduced documentation. Information of third parties may be subject to additional restrictions.

Buyers and others who are developing systems that incorporate Realtek products (collectively, "Customers") understand and agree that Customers remain responsible for using their independent analysis, evaluation and judgment in designing their applications and that Customers have full and exclusive responsibility to assure the safety of Customers' applications and compliance of their applications (and of all Realtek products used in or for Customers' applications) with all applicable regulations, laws and other applicable requirements. Designer represents that, with respect to their applications, Customer has all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. Customer agrees that prior to using or distributing any applications that include Realtek products, Customer will thoroughly test such applications and the functionality of such Realtek products as used in such applications.

Realtek's provision of technical, application or other design advice, quality characterization, reliability data or other services or information, including, but not limited to, reference designs and materials relating to evaluation kits, (collectively, "Resources") are intended to assist designers who are developing applications that incorporate Realtek products; by downloading, accessing or using Realtek's Resources in any way, Customer (individually or, if Customer is acting on behalf of a company, Customer's company) agrees to use any particular Realtek Resources solely for this purpose and subject to the terms of this Notice.

Realtek's provision of Realtek Resources does not expand or otherwise alter Realtek's applicable published warranties or warranty disclaimers for Realtek's products, and no additional obligations or liabilities arise from Realtek providing such Realtek Resources. Realtek reserves the right to make corrections, enhancements, improvements and other changes to its Realtek Resources. Realtek has not conducted any testing other than that specifically described in the published documentation for a particular Realtek Resource.

Customer is authorized to use, copy and modify any individual Realtek Resource only in connection with the development of applications that include the Realtek product(s) identified in such Realtek Resource. No other license, express or implied, by estoppel or otherwise to any other Realtek intellectual property right, and no license to any technology or intellectual property right of Realtek or any third party is granted herein, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which Realtek products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of Realtek Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from Realtek under the patents or other Realtek's intellectual property.

Realtek's Resources are provided "as is" and with all faults. Realtek disclaims all other warranties or representations, express or implied, regarding resources or use thereof, including but not limited to accuracy or completeness, title, any epidemic failure warranty and any implied warranties of merchantability, fitness for a particular purpose, and non-infringement of any third party intellectual property rights.

Realtek shall not be liable for and shall not defend or indemnify Customer against any claim, including but not limited to any infringement claim that related to or is based on any combination of products even if described in Realtek Resources or otherwise. In no event shall Realtek be liable for any actual, direct, special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of Realtek's Resources or use thereof, and regardless of whether Realtek has been advised of the possibility of such damages. Realtek is not responsible for any failure to meet such industry standard requirements.

Where Realtek specifically promotes products as facilitating functional safety or as compliant with industry functional safety standards, such products are intended to help enable customers to design and create their own applications that meet applicable functional safety standards and requirements. Using products in an application does not by itself establish any safety features in the application. Customers must ensure compliance with safety-related requirements and standards applicable to their applications. Designer may not use any Realtek products in life-critical medical equipment unless authorized officers of the parties have executed a special contract specifically governing such use. Life-critical medical equipment is medical equipment where failure of such equipment would cause serious bodily injury or death. Such equipment includes, without limitation, all medical devices identified by the U.S.FDA as Class III devices and equivalent classifications outside the U.S.

Customers agree that it has the necessary expertise to select the product with the appropriate qualification designation for their applications and that proper product selection is at Customers' own risk. Customers are solely responsible for compliance with all legal and regulatory requirements in connection with such selection.

Customer will fully indemnify Realtek and its representatives against any damages, costs, losses, and/or liabilities arising out of Designer's non-compliance with the terms and provisions of this Notice.

TRADEMARKS

Realtek is a trademark of Realtek Semiconductor Corporation. Other names mentioned in this document are trademarks/registered trademarks of their respective owners.

USING THIS DOCUMENT

Though every effort has been made to ensure that this document is current and accurate, more information may have become available subsequent to the production of this guide.

All information provided in this document is subject to legal disclaimers.

© REALTEK 2022. All rights reserved.

Table of Contents

TABLE OF CONTENTS **4**

1 DOCUMENT INFORMATION **5**

1.1 Revision History 5

2 OVERVIEW **6**

3 HARDWARE DESCRIPTION **7**

3.1 Datasheet 7
3.2 Schematic 7
3.3 Key Components 8

4 CONFIGURE AWS IOT CORE **9**

4.1 Set up Your AWS Account and Permissions 9
4.2 Create a New Device 9
4.3 Create a Policy 13
4.4 Attach Policy 14

5 CONFIGURE FREERTOS-LTS ON AMEBAPRO2 ..... **17**

5.1 Download FreeRTOS-LTS Library Source Code from Github 17
5.2 Get Broker Endpoint by AWS IoT Core 17
5.3 Get Thing Name 18
5.4 Setup IoT Core Information for Demo 18

6 SET UP YOUR DEVELOPMENT ENVIRONMENT **21**

6.1 CMake install 21
6.2 Toolchains 21
6.3 Build Project 21
6.4 Establishing a Serial Connection 21
6.5 Image Download 22

7 MQTT DEMO **24**

7.1 Run MQTT Demo 24
7.2 Monitoring MQTT Messages on the Cloud 25

8 OTA DEMO **27**

8.1 OTA Update Prerequisites 27
8.2 Set the App Version to Image File 27
8.3 Custom-Signed Image and Signature 28
8.4 How to Trigger a Custom Signed OTA Job in Amazon AWS IOT Core 28
8.5 Run OTA Demo 33

9 TROUBLESHOOTING **36**

9.1 ERROR: Invalid Key 36
9.2 Failed to Establish New MQTT Connection 36
9.3 TLS_Connect Fail 36
9.4 OTA Signature Check Fail 37

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

1 Document Information

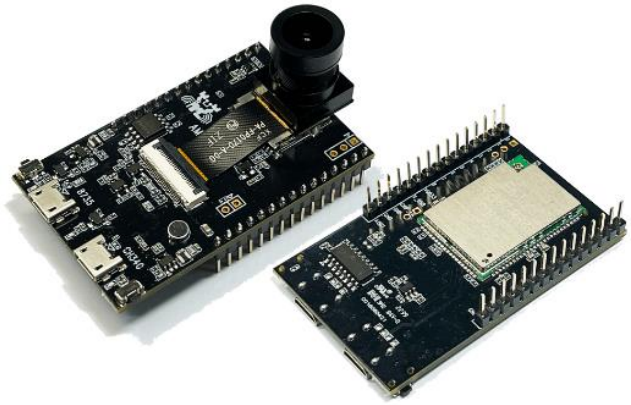
1.1 Revision History

Version	Release Date	Description of change
1.3.0	2022.10	1. First version for public release
1.3.1	2022.12	1. Add OTA demo in detail
1.3.2	2022.12	1. Add a section to describe hardware

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

2 Overview

AmebaPro2 board is an easy-to-program platform for developing all kind of IoT applications. AmebaPro2 is equipped with various peripheral interfaces, including WiFi, BLE, GPIO INT, I2C, UART, SPI, PWM, and ADC. Through these interfaces, AmebaPro2 can connect with electronic components such as LED, switches, manometer, hygrometer, PM2.5 dust sensors …etc. Besides, AmebaPro2 has three key features: Audio codec, Video codec and NN (build in NPU for AIoT).



MCU

Part Number: RTL8735B

32-bit Arm v8M, up to 500MHz

MEMORY

768KB ROM

512KB RAM

Supports MCM embedded DDR2/DDR3L memory up to 128MB

External Flash up to 64MB

KEY FEATURES

Integrated 802.11 a/b/g/n Wi-Fi, 2.4GHz/5GHz

Bluetooth Low Energy (BLE) 4.2

Integrated Intelligent Engine @ 0.4 TOPS

Ethernet Interface

USB Host/Device

SD Host

ISP

Audio Codec

H.264/H.265

Secure Boot

Crypto Engine

OTHER FEATURES

2 SPI interfaces

1 I2C interface

8 PWM interfaces

3 UART interfaces

3 ADC interfaces

2 GDMA interfaces

Max 23 GPIO

AmebaPro2 uses Micro USB to supply power, which is common in many smart devices.

Ameba Demo board home page: https://www.amebaiot.com/en/amebapro2/

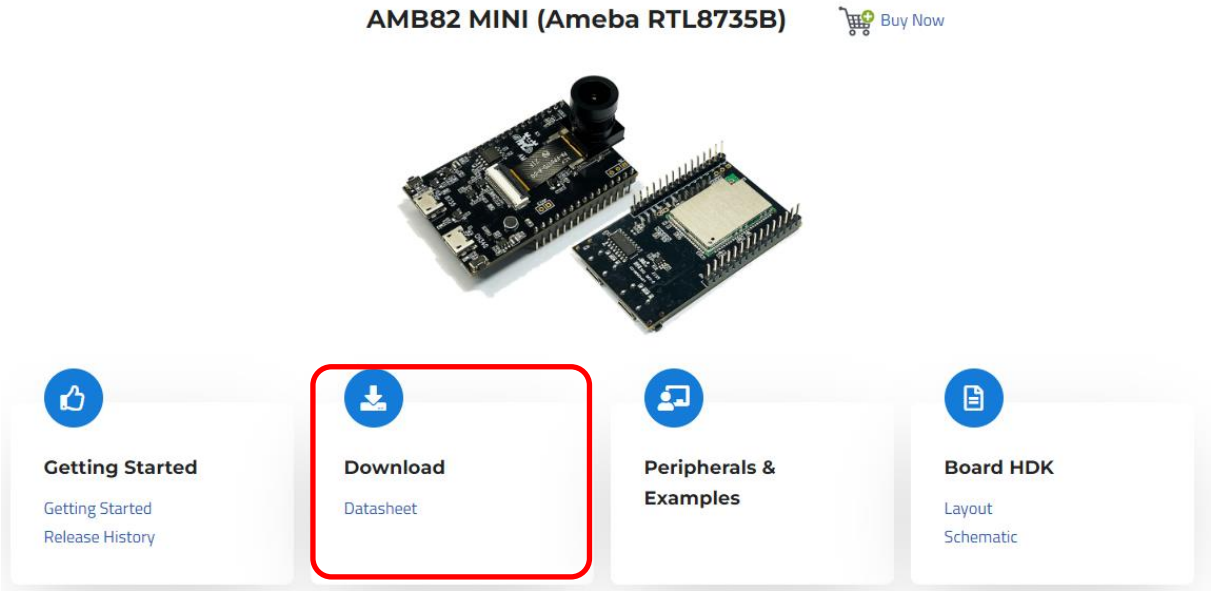
All information provided in this document is subject to legal disclaimers.

© REALTEK 2022. All rights reserved.

3 Hardware Description

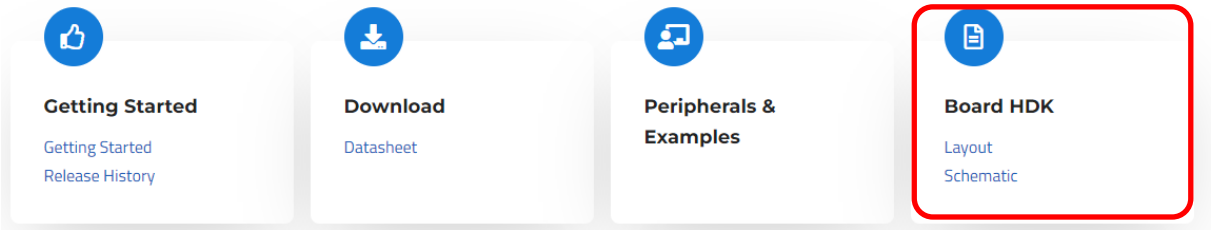
3.1 Datasheet

User can download the datasheet from ameba official website – AmebaPro2 datasheet



3.2 Schematic

The PCB schematic can be downloaded from ameba official website – board schematic



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

3.3 Key Components

The PCB layout of AmebaPro2 is shown in Fig 3-1.

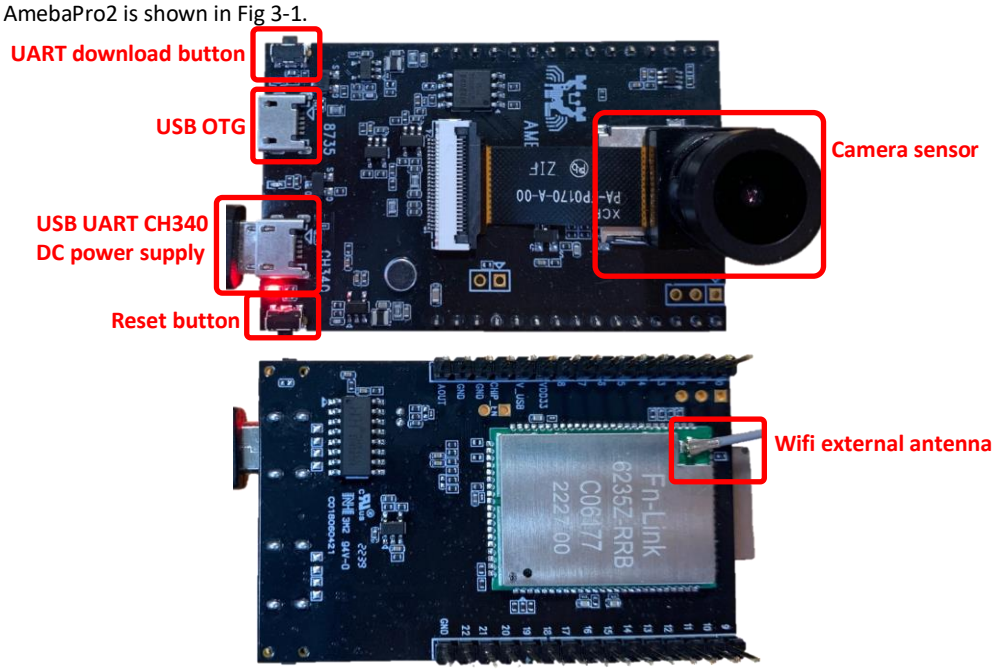


Fig 3-1 Demo board – PCB layout (2D)

Please refer to the following Fig 3-2 for the pin diagram and functions.

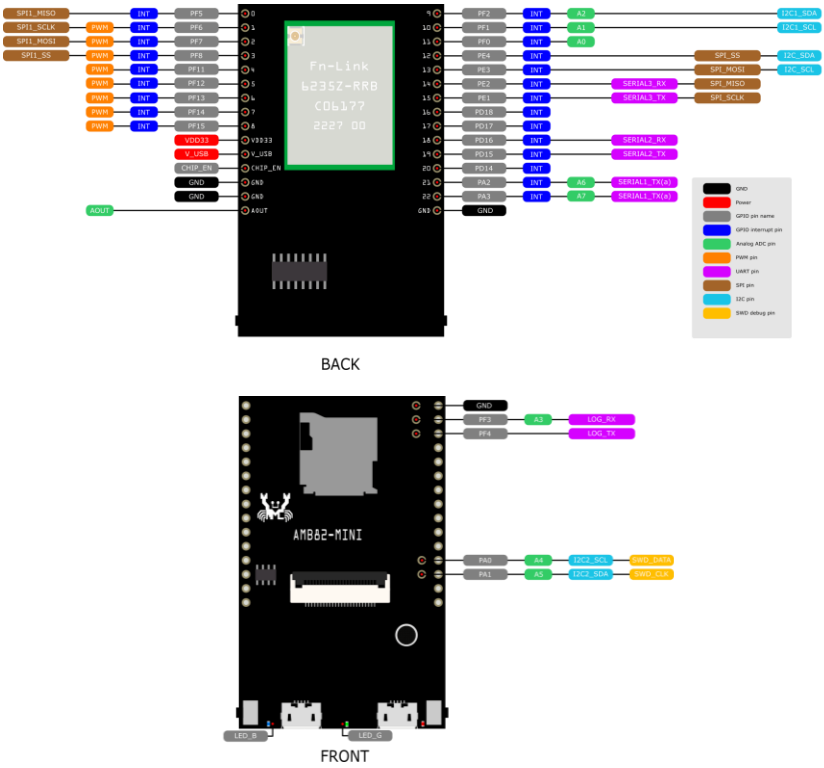


Fig 3-2 pin diagram

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

4 Configure AWS IoT Core

4.1 Set up Your AWS Account and Permissions

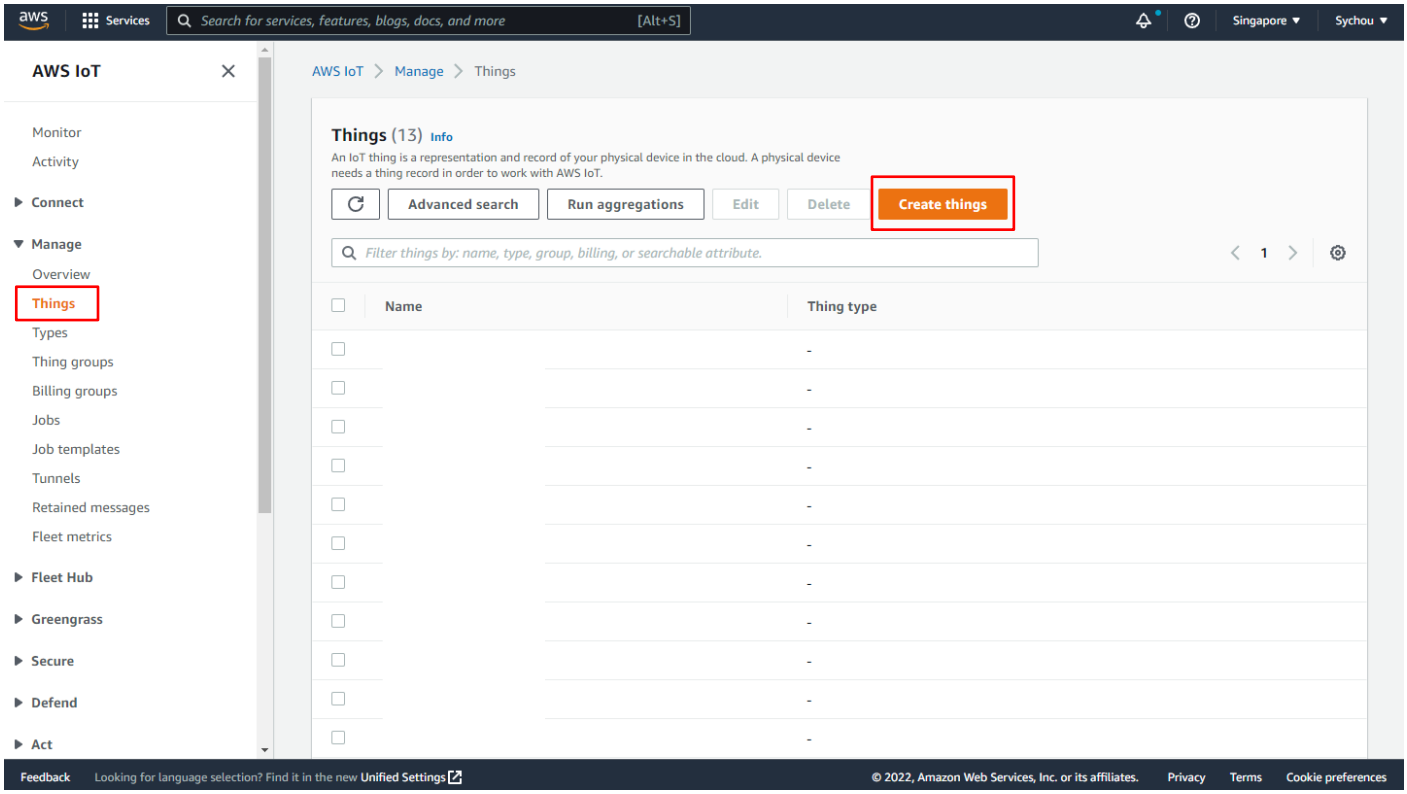
Refer to the instructions at Set up your AWS Account https://docs.aws.amazon.com/iot/latest/developerguide/setting-up.html. Follow the steps outlined in these sections to create your account and a user and get started:

- Sign up for an AWS account
- Create a user and grant permissions
- Open the AWS IoT console

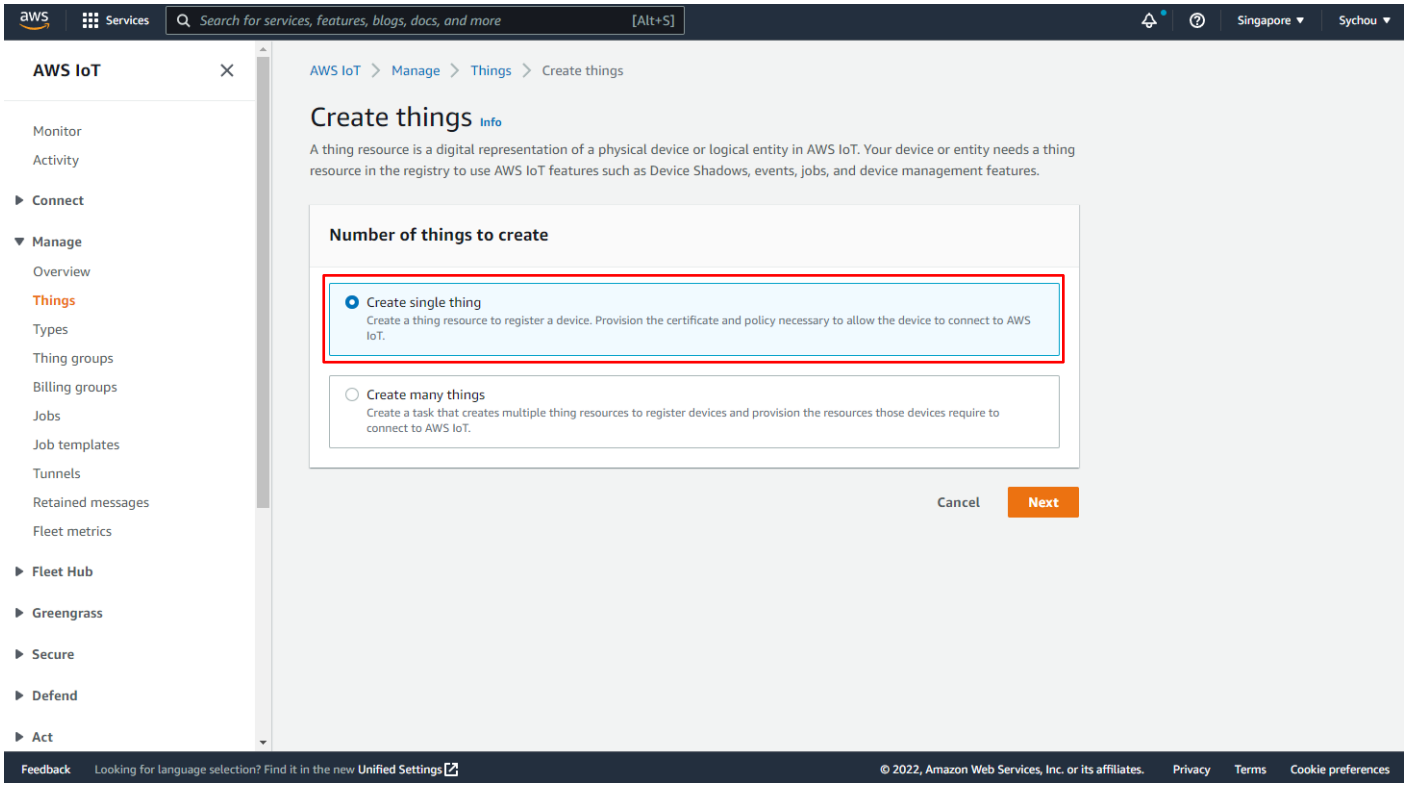
Please pay special attention to the Notes in AWS webpage.

4.2 Create a New Device

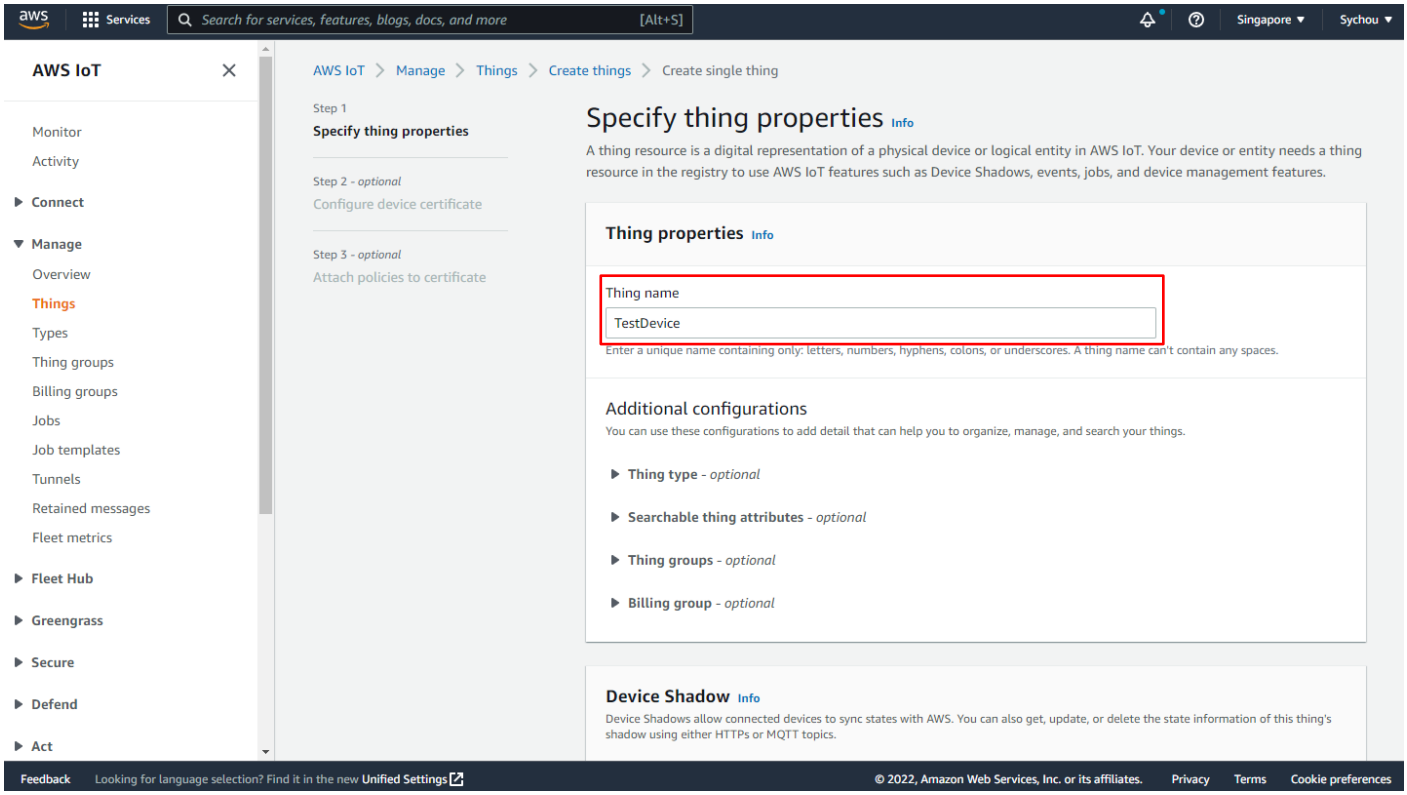
To create a new device, navigate to Manage -> Things in the left-hand navigation menu. Then click “Create things”.



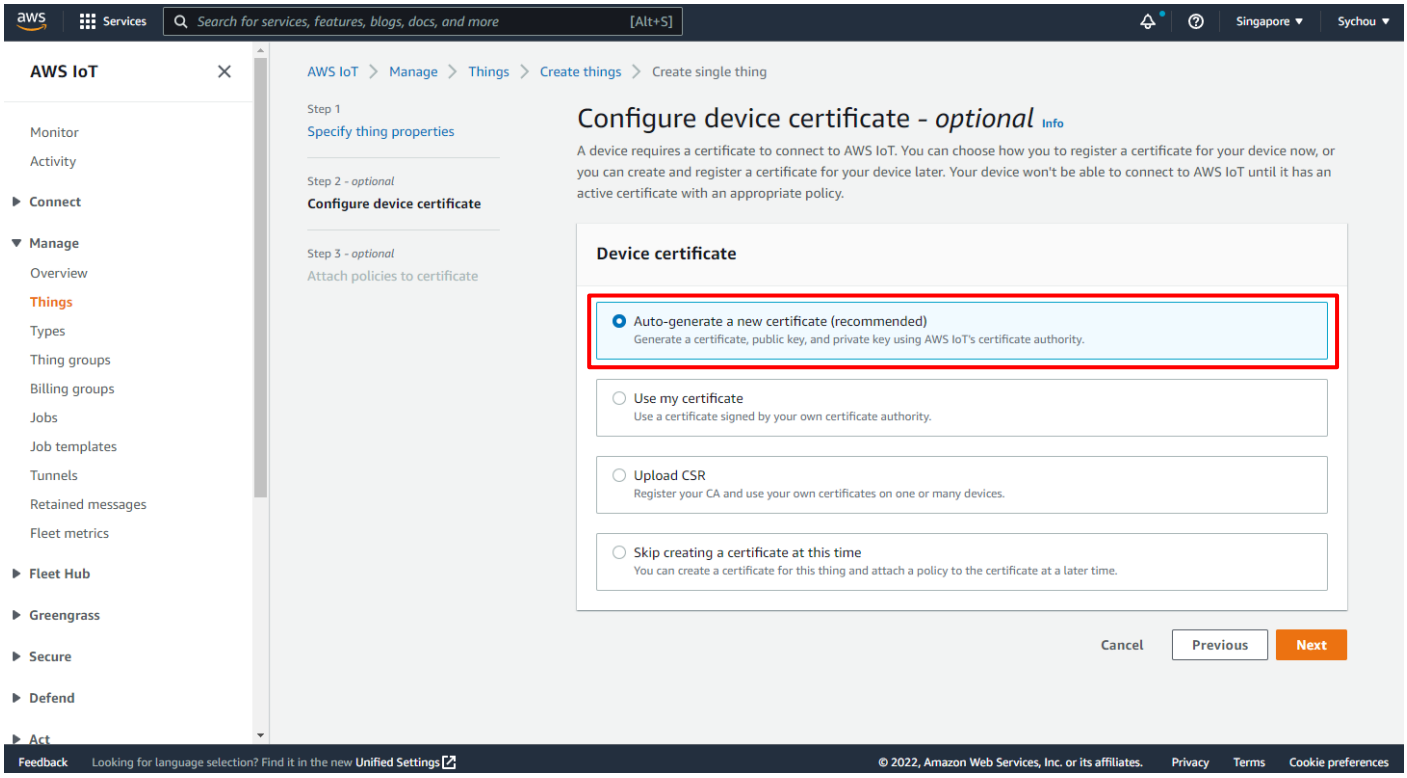
All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.



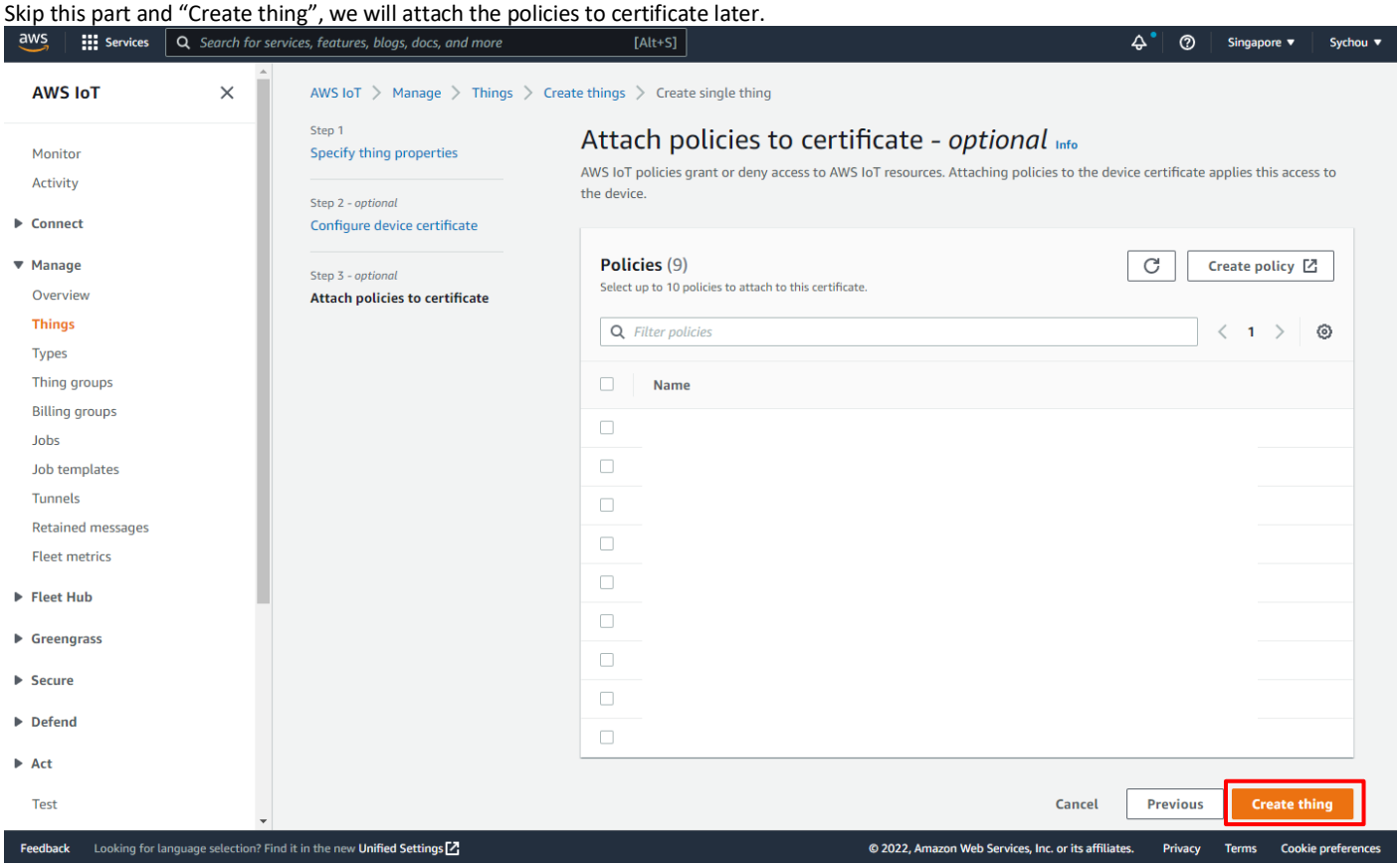
Then, name the new device. This example uses the name TestDevice.



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

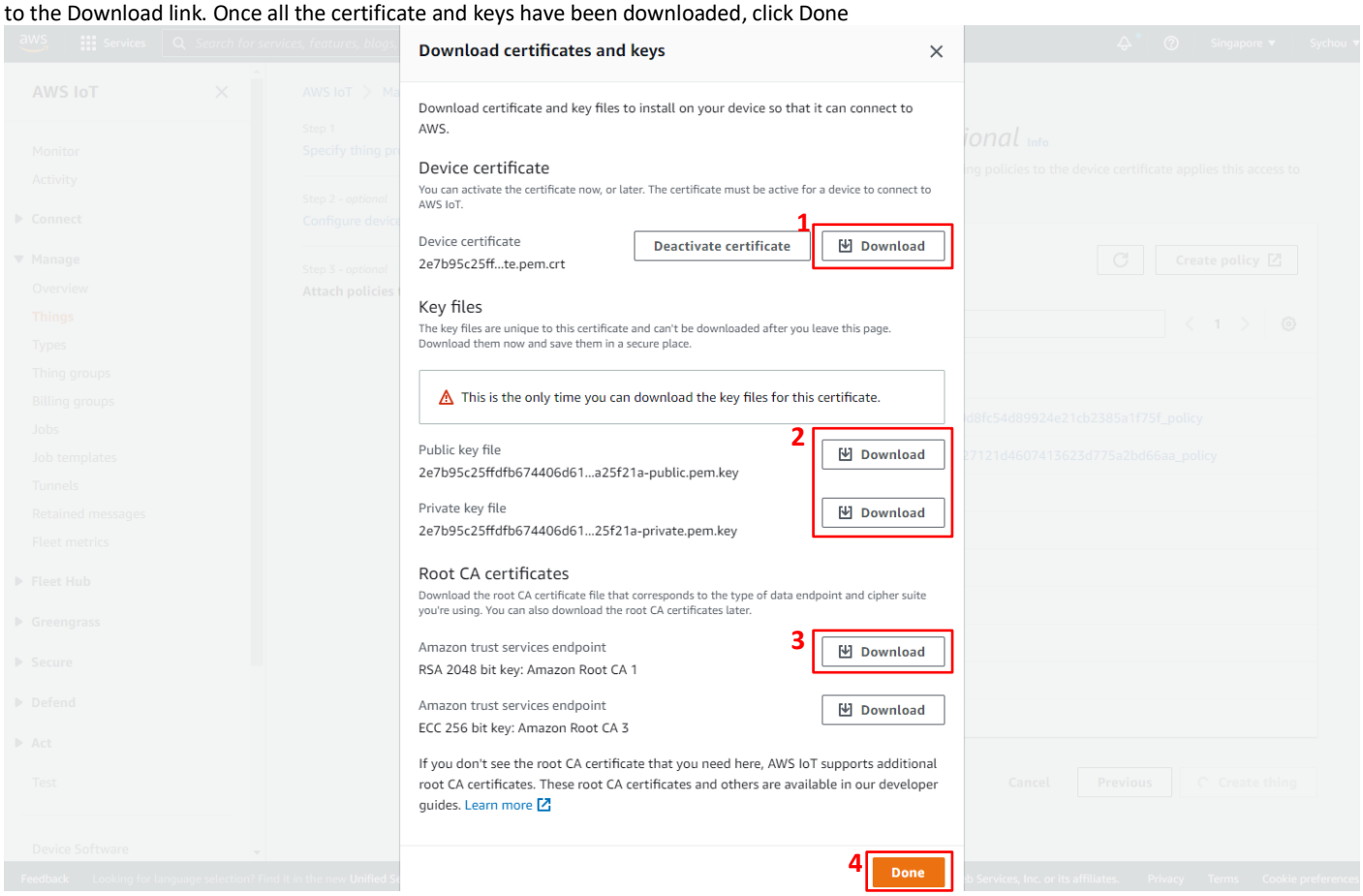


Skip this part and “Create thing”, we will attach the policies to certificate later.



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

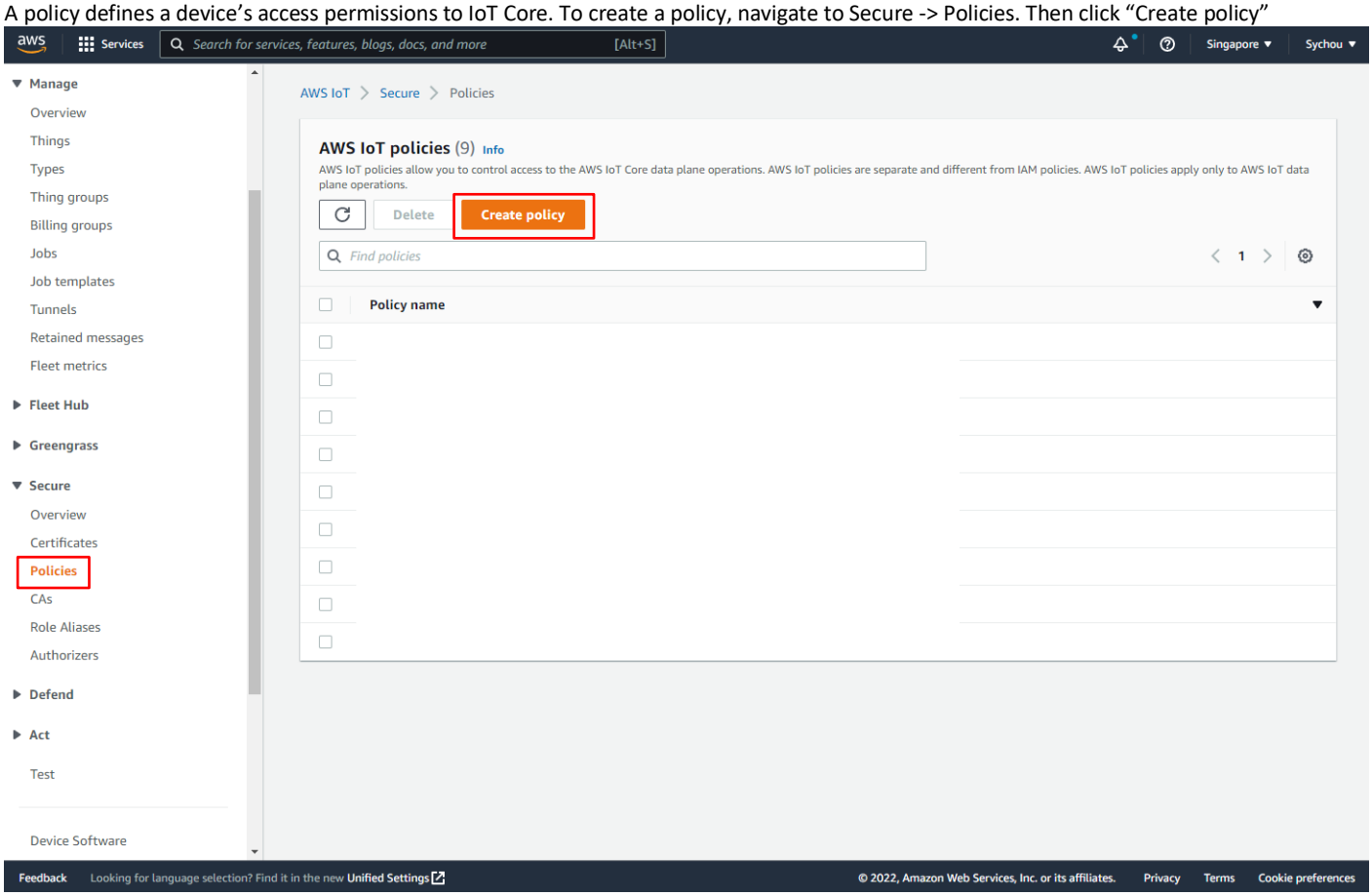
Download the certificate, public key, and private key for the device by clicking Download. Next, download the root CA for AWS IoT by clicking to the Download link. Once all the certificate and keys have been downloaded, click Done



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

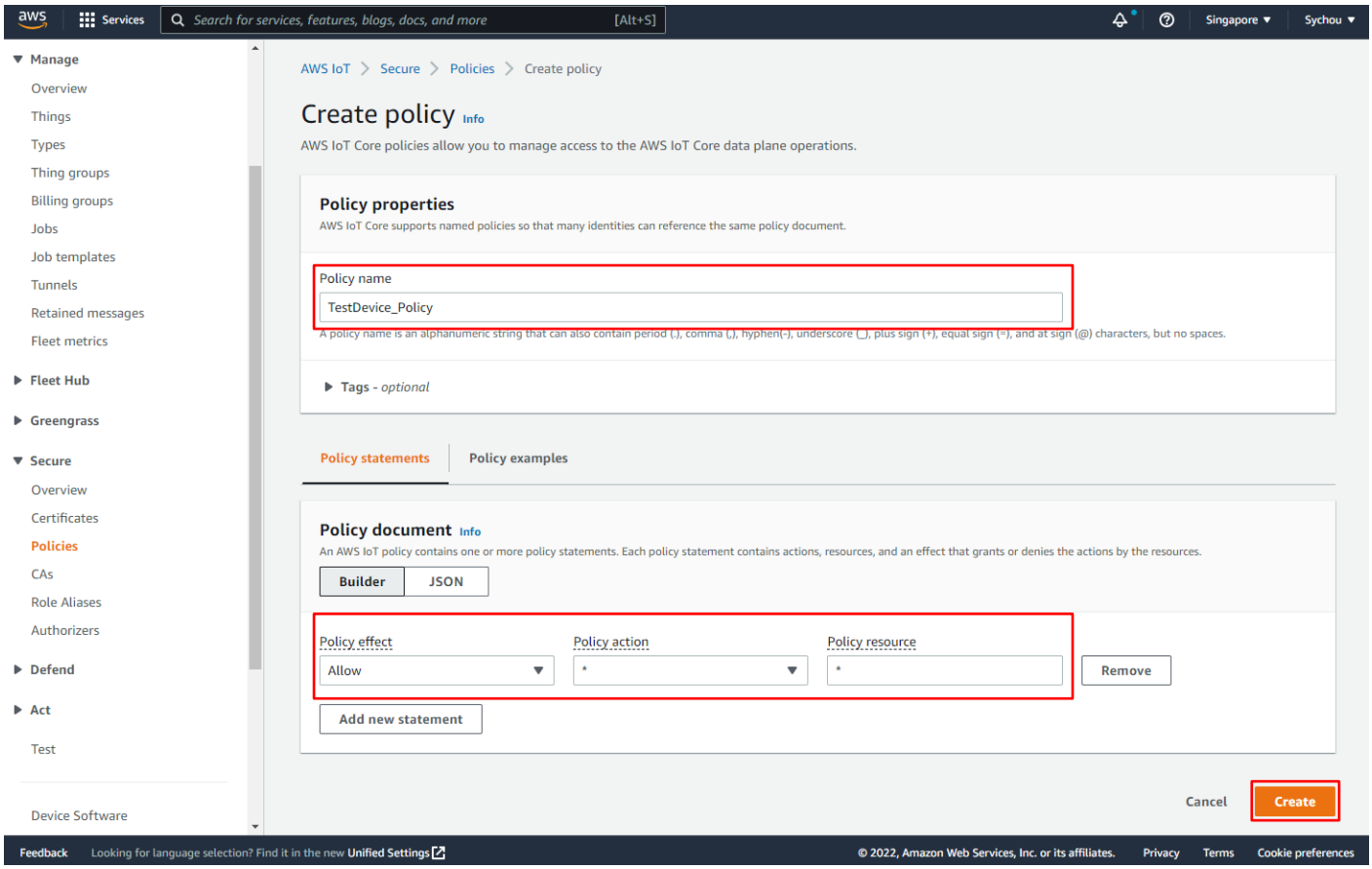
4.3 Create a Policy

A policy defines a device's access permissions to IoT Core. To create a policy, navigate to Secure -> Policies. Then click "Create policy"



Note: this policy grants unrestricted access for all iot operations, and is to be used only in a development environment. For non-dev environments, all devices in your fleet must have credentials with privileges that authorize intended actions only, which include (but not limited to) AWS IoT MQTT actions such as publishing messages or subscribing to topics with specific scope and context. The specific permission policies can vary for your use cases. Identify the permission policies that best meet your business and security requirements.
For sample policies, refer to https://docs.aws.amazon.com/iot/latest/developerguide/example-iot-policies.html.
Also refer to https://docs.aws.amazon.com/iot/latest/developerguide/security-best-practices.html

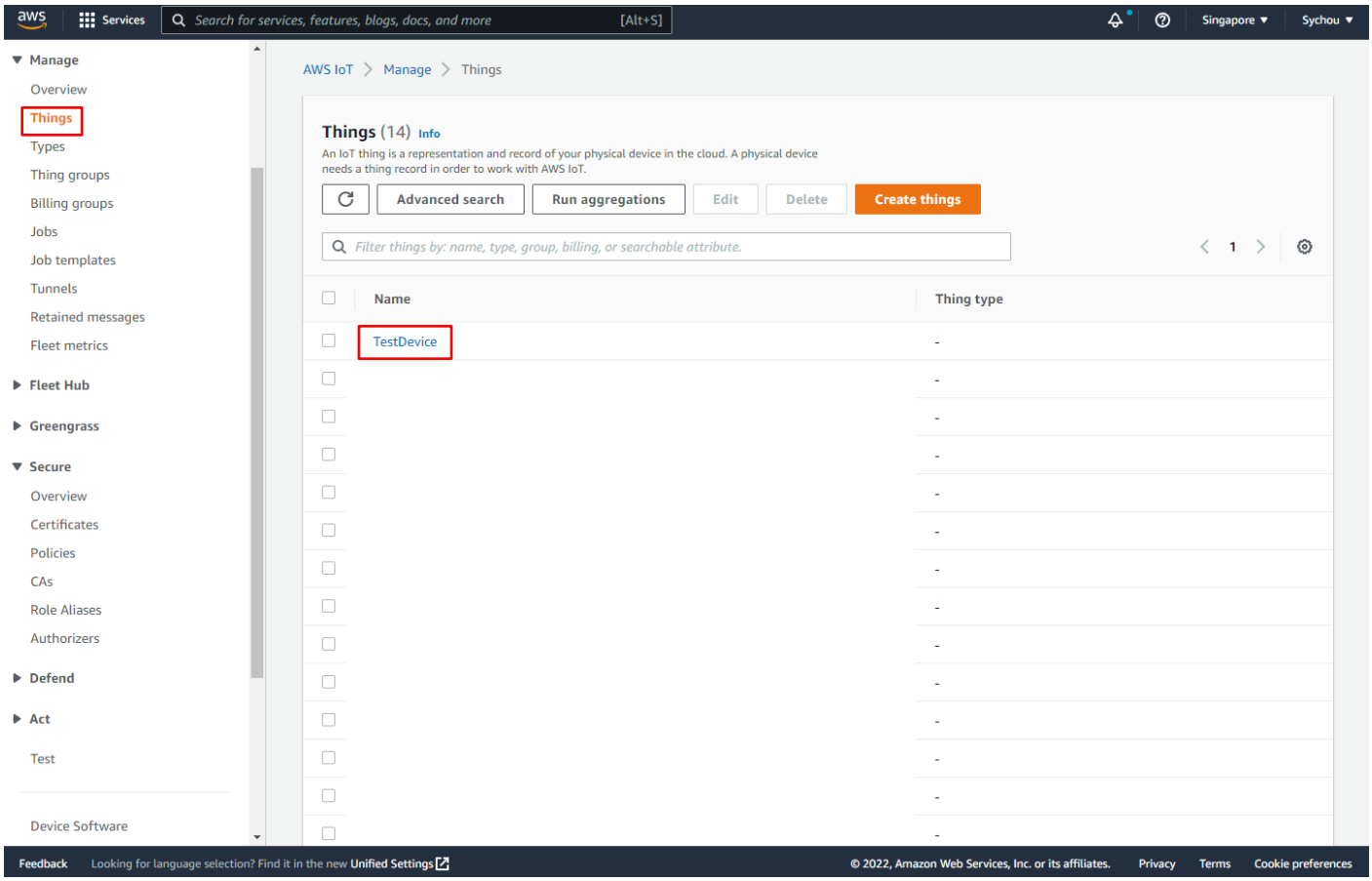
All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.



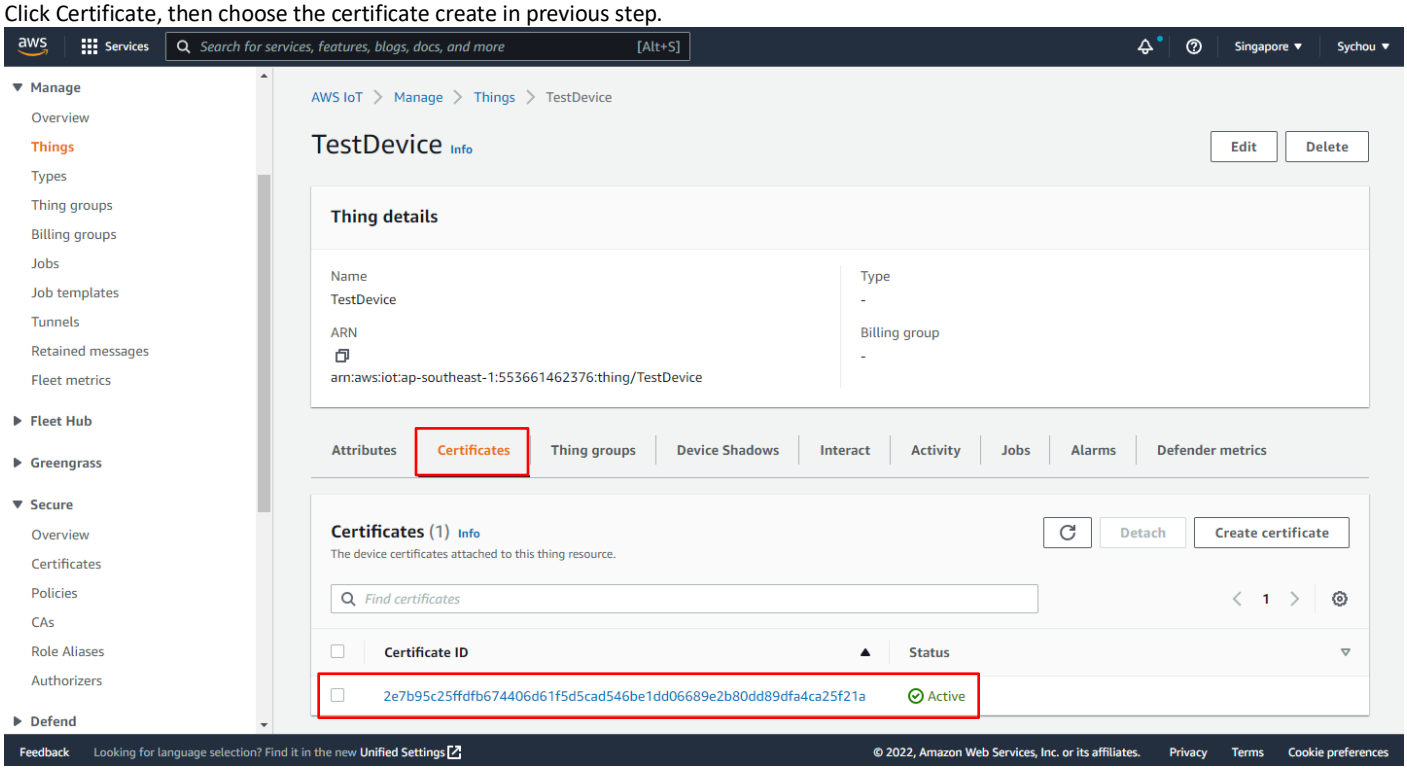
4.4 Attach Policy

The last step to configuring the device is attaching a policy. To attach a policy to new device, navigate to Manage -> Things. Then click on the device which was created.

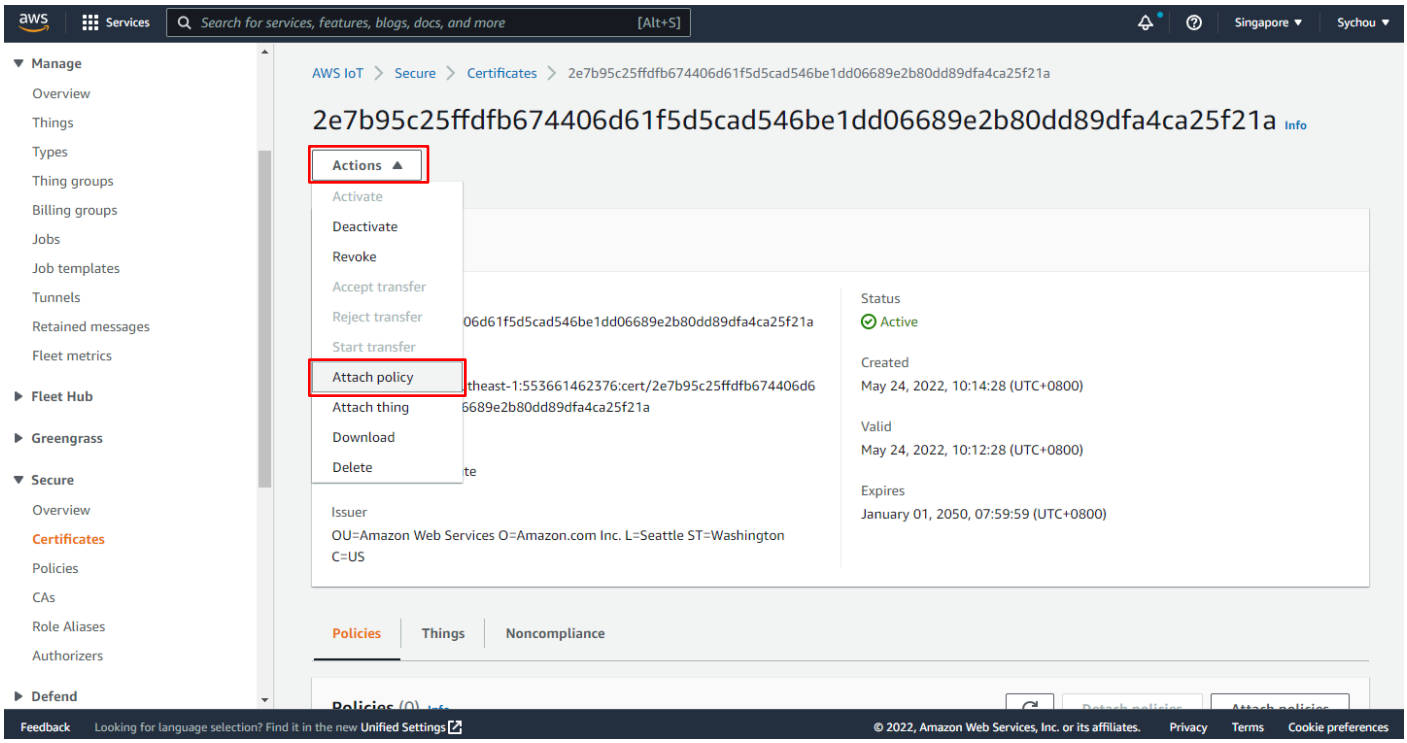
All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

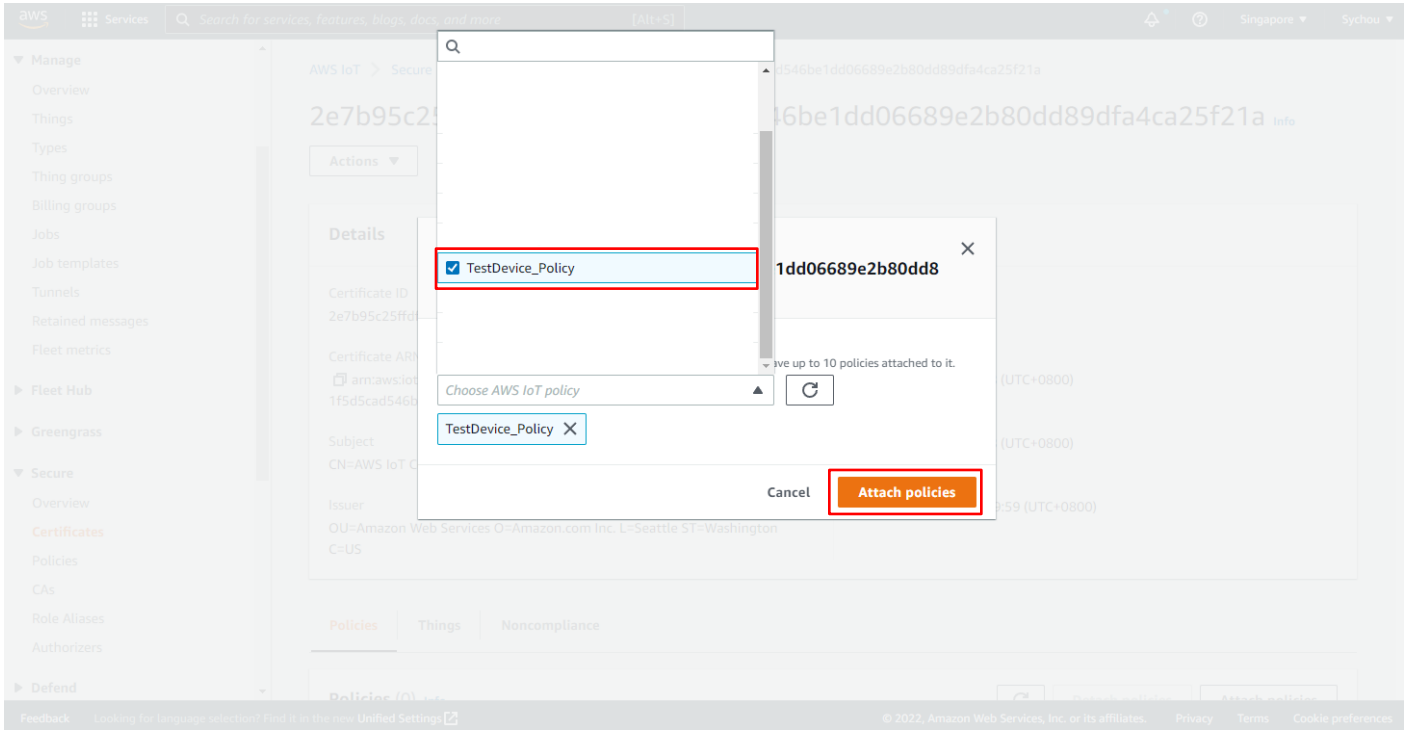


Click Certificate, then choose the certificate create in previous step.



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.





All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

5 Configure FreeRTOS-LTS on AmebaPro2

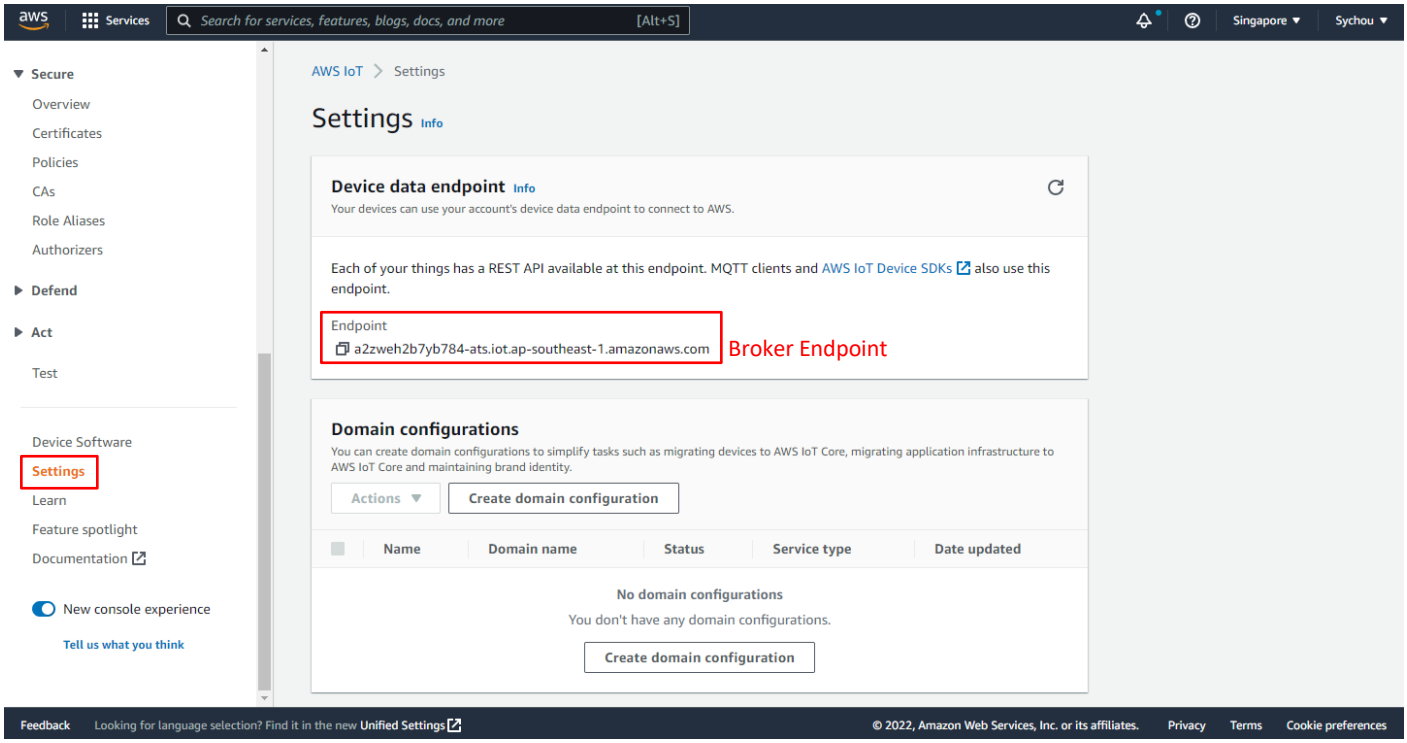
5.1 Download FreeRTOS-LTS Library Source Code from Github

Open source link: https://github.com/ambiot/amazon-freertos/tree/amebaPro2-9.x-202012-LTS
branch: **amebaPro2-9.x-202012-LTS**

Go to "AmebaPro2_SDK/project/realtek_amebapro2_v0_example/src":

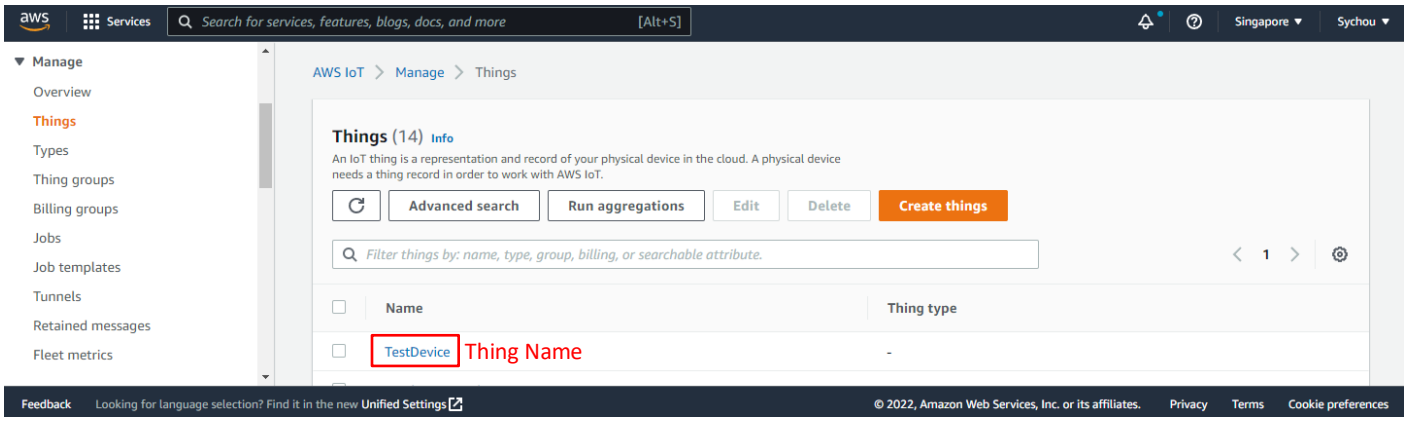
$ cd project/realtek_amebapro2_v0_example/src
$ git clone --recurse-submodules -b amebaPro2-9.x-202012-LTS https://github.com/ambiot/amazon-freertos.git aws_iot_freertos_lts

5.2 Get Broker Endpoint by AWS IoT Core



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

5.3 Get Thing Name



5.4 Setup IoT Core Information for Demo

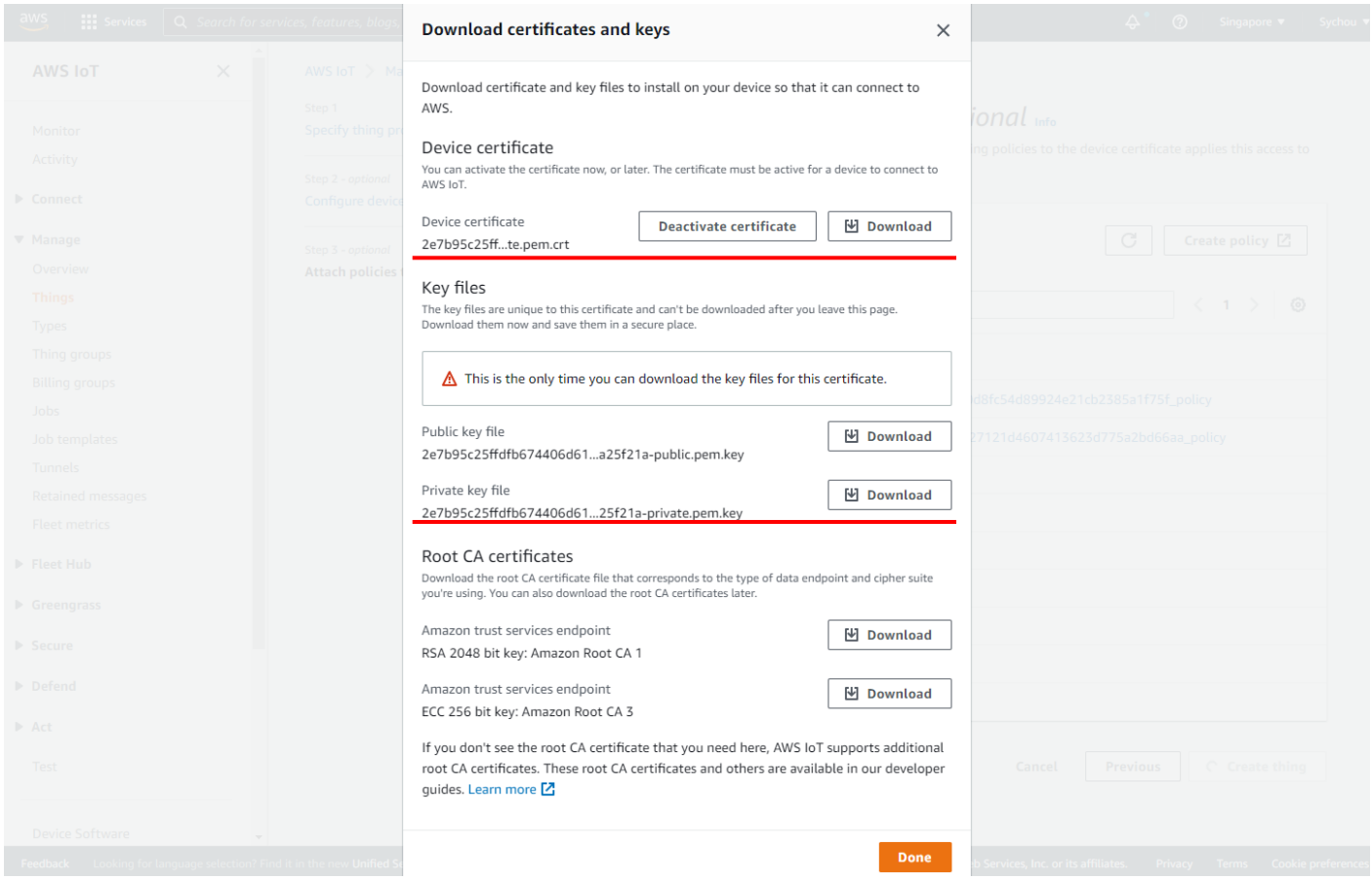
Setup BROKER_ENDPOINT, THING_NAME, WIFI_SSID, PASSWORD in
“project/realtek_amebapro2_v0_example/src/aws_iot_freertos_lts/demos/include/aws_clientcredential.h”

```
#define clientcredentialMQTT_BROKER_ENDPOINT        "xxxxxxxxxxxxxx.amazonaws.com"

/*
 * @brief Host name.
 *
 * @todo Set this to the unique name of your IoT Thing.
 */
#define clientcredentialIOT_THING_NAME              "TestDevice"

/*
 * @brief Port number the MQTT broker is using.
 */
#define clientcredentialMQTT_BROKER_PORT            8883

/*
 * @brief Port number the Green Grass Discovery use for JSON retrieval from cloud is using.
 */
#define clientcredentialGREENGRASS_DISCOVERY_PORT   8443

/*
 * @brief Wi-Fi network to join.
 *
 * @todo If you are using Wi-Fi, set this to your network name.
 */
#define clientcredentialWIFI_SSID                   "TestAP"

/*
 * @brief Password needed to join Wi-Fi network.
 * @todo If you are using WPA, set this to your network password.
 */
#define clientcredentialWIFI_PASSWORD               "password"

/*
 * @brief Wi-Fi network security type.
 *
 * @see WIFISecurity_t.
 *
 * @note Possible values are eWiFiSecurityOpen, eWiFiSecurityWEP, eWiFiSecurityWPA,
 * eWiFiSecurityWPA2 (depending on the support of your device Wi-Fi radio).
 */
#define clientcredentialWIFI_SECURITY               eWiFiSecurityWPA2

#endif /* ifndef __AWS_CLIENTCREDENTIAL__H__ */
```


All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

5.4.1 Setup Thing's Private Key and Certificate

Fill keyCLIENT_CERTIFICATE_PEM and keyCLIENT_PRIVATE_KEY_PEM in "project/realtek_amebapro2_v0_example/src/aws_iot_freertos_lts/demos/include/aws_clientcredential_keys.h" by xxxxxxxx-certifiacte.pem and xxxxxxxx-private.pem.key.



It can be done by the script PEM-to-C-string.py provided by AWS. It can be downloaded from https://github.com/aws/amazon-freertos/tree/main/tools/certificate_configuration.

```
$ python3 PEM-to-C-string.py --cert-file 31ba70aa1e99443b373438e1f71ec81a207e7d88964215e7d6e63eadde85b3ef-certificate.pem.crt
========== Formatted 31ba70aa1e99443b373438e1f71ec81a207e7d88964215e7d6e63eadde85b3ef-certificate.pem.crt ==========

#define keyCLIENT_CERTIFICATE_PEM \
"-----BEGIN CERTIFICATE-----\n" \
"MIIDWjCCAkKgAwIBAgIVAIju/DZ7pt9mzL46E2ALViDDaTsTMA0GCSqGSIb3DQEB\n" \
"CwUAME0xSzBJBgNVBAsMQkFtYXpvbiBXZWIgU2VydmljZXMgTz1BbWF6b24uY29t\n" \
"IEluYy4gTD1TZWF0dGx1IFNUPVdhc2hpbmd0b24gQz1VUzAeFw0yMjA4MjYxMDIw\n" \
"MTJaFw0OOTEyMzEyMzU5NTlaMB4xHDAaBgNVBAMMEOFXUyBJb1QgQ2VydGlmaWNh\n" \
"dGUwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC2nc04SU8SIMyZaoc0\n" \
"LtgvJ0uaHuVR8QOT2Fq83d7XuUGppSQKNAsgER1dseV4m8/NYOECt04+6AJRj8OD\n" \
"qZXG+pz5TcmS1eAqDkh+7aBwwPhCy6g7aVCzK6dmKCfVL1vuazgjzfjXHTjZDpdX\n" \
"nfjayi7u5ZJQEf43a0c1UqLbm8QeCEjaGkIvPHgQQ6fosURSaaze7NQwvRCb1IRF\n" \
"5398+GdzplXIVcotf8yEP+oihyY757rpJvrJPZXLNpPO6yccoDIGMn6W1GsVhhbY\n" \
"U6f1TQ7c33ya/TwWu53DkMzyvtVQHqZF17gBn2SMLf2zaaPbvod2wHh9MG5vYiOc\n" \
"oofDAgMBAAGjYDBeMB8GA1UdIwQYMBaAFHxSAJH8s8IxJfK5kpGO1gQ+qswQMB0G\n" \
"A1UdDgQWBBQSxHP8vPR6VLHXVOdk28Q3Jma1fzAMBgNVHRMBAf8EAjAAMA4GA1Ud\n" \
"DwEB/wQEAwIHgDANBgkqhkiG9w0BAQsFAAOCAQEARCCbonojgoIC9iolvtCmFz3r\n" \
"TsnuIW+8Psl1pxhsBwuTpUlFObYZardXkkMyFoZz9hUcbr6OftP90PkpbNPL/A9x\n" \
"ZuY63Ub4fRTunmKatYOueWwOkdCXDpcpLpNdVjWttnSEHGD7KxIY7Q1sY5vzKgbu\n" \
"JqMuPbYRb1DV+9FVtg6ngZj6RdCykRiTjb/BXtkzdYFQ1/nFR8Q4QuIKkFKplZjV\n" \
"4NkWPZ9gzgMgK/W34LMITALnpjO4ruio5wWN+cbPmlvO6ZT+G/8YkSPnVbObahXy\n" \
"fwgojljcV7bQtPgUgM3GLUhGgs+5O+BjFneMLIJYCFHkayCmo94cJgRIg6BgUQ==\n" \
"-----END CERTIFICATE-----\n"
```


All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

Final aws_clientcredential_keys.h overview.

```
/*
 * @brief PEM-encoded client certificate.
 *
 * @todo If you are running one of the FreeRTOS demo projects, set this
 * to the certificate that will be used for TLS client authentication.
 *
 * @note Must include the PEM header and footer:
 * "-----BEGIN CERTIFICATE-----\n"\
 * "...base64 data...\n"\
 * "-----END CERTIFICATE-----\n"
 */
#define keyCLIENT_CERTIFICATE_PEM \
"-----BEGIN CERTIFICATE-----\n"\
"MIIDWjCCAkKgAwIBAgIVAIDLSSoG+EARSbBprT4Im1uu8j2vMA0GCSqGSIb3DQEB\n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"k5+NsBroU/YdvOUmzKn6XfI4nX4hLQJ2TbhAT8aq1ounGk6ZGqCbxt4mg5bB0w==\n"\
"-----END CERTIFICATE-----"
```

```
/*
 * @brief PEM-encoded client private key.
 *
 * @todo If you are running one of the FreeRTOS demo projects, set this
 * to the private key that will be used for TLS client authentication.
 *
 * @note Must include the PEM header and footer:
 * "-----BEGIN RSA PRIVATE KEY-----\n"\
 * "...base64 data...\n"\
 * "-----END RSA PRIVATE KEY-----\n"
 */
#define keyCLIENT_PRIVATE_KEY_PEM \
"-----BEGIN RSA PRIVATE KEY-----\n"\
"MIIEpAIBAAKCAQEAwop96WNucGebARFjD8O+CLsqcBNn/AHyhEcozLZC8qoECUOn\n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"                                                                \n"\
"pOWEuLUuz2FAv1noAbN/6OQ8H/PTOAFJT/ghA04GnIUF0kjSzY60ehS2mVp6neP+\n"\
"AZjzZ6QJYlb5/PFz9oES448kpyaAoS2ke86+R4r4YOMBK+I5RVbfSQ==\n"\
"-----END RSA PRIVATE KEY-----\n"
```

5.4.2 Enable FreeRTOS Demo on AmebaPro2

For example, if you would like to run MQTT mutual authentication demo, please find aws_demo_config.h in "project/realtek_amebapro2_v0_example/src/aws_iot_freertos_lts/vendors/realtek/boards/amebaPro2/aws_demos/config_files/" and enable **CONFIG_CORE_MQTT_MUTUAL_AUTH_DEMO_ENABLED**

```
//#define CONFIG_CORE_HTTP_MUTUAL_AUTH_DEMO_ENABLED
#define CONFIG_CORE_MQTT_MUTUAL_AUTH_DEMO_ENABLED
//#define CONFIG_DEVICE_SHADOW_DEMO_ENABLED
//#define CONFIG_JOBS_DEMO_ENABLED
```

Now you can start to compile AmebaPro2 Amazon FreeRTOS project!

All information provided in this document is subject to legal disclaimers.

© REALTEK 2022. All rights reserved.

6 Set up your Development Environment

6.1 CMake install

Install cmake by using terminal. If the installation is successful, you can get the version by “cmake -version”.

$ sudo apt install cmake

6.2 Toolchains

Extract the toolchain file (the toolchain file may provide in tools folder):

$ cd <AmebaPro2_SDK>/tools
$ cat asdk-10.3.0-linux-newlib-build-3633-x86_64.tar.bz2.* | tar jxvf -

Note: If using windows, you can build the project by WSL(Windows Subsystem for Linux)

Add toolchain to PATH:

$ export PATH=$PATH:<AmebaPro2_SDK>/tools/asdk-10.3.0/linux/newlib/bin

Note: you can add it to ~/.bash_profile

6.3 Build Project

User can follow the following steps to build the project with aws demo:

Step 1. Open linux terminal and enter the project location: project/realtek_amebapro2_v0_example/GCC-RELEASE/.
Step 2. Create folder “build” and enter “build” folder.
Step 3. Run “cmake .. -G"Unix Makefiles" -DCMAKE_TOOLCHAIN_FILE=../toolchain.cmake -DEXAMPLE=amazon_freertos” to create makefile.
Step 4. Run “cmake --build . --target flash -j4” to build and generate flash binary.

In summary, user can run following commands with option “-DEXAMPLE=amazon_freertos” to build the image

$ cd project/realtek_amebapro2_v0_example/GCC-RELEASE
$ mkdir build
$ cd build
$ cmake .. -G"Unix Makefiles" -DCMAKE_TOOLCHAIN_FILE=../toolchain.cmake -DEXAMPLE=amazon_freertos
$ cmake --build . --target flash -j4

After successfully build, there should be an image file **flash_ntz.bin** located in “build/” directory.

Note: If there is some permission issues, you can do “chmod -R 777 <PATH_TO_YOUR_SDK>”

6.4 Establishing a Serial Connection

For the log UART setting:

1. Connect AmebaPro2 to the computer via Micro USB as shown in Fig 6-1
2. After connecting to PC, create a console session by console tools (like tera term, MoBaxterm) with baud rate 115200
3. Press the reset button, then you can see the log from EVB

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

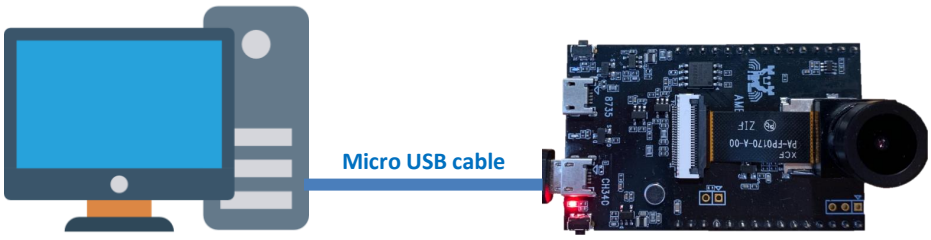


Fig 6-1 Demo board – log UART

The default image boot up console log shown as Fig 6-2:

```
== Rtl8735b IoT Platform ==
Chip VID: 0, Ver: 0
ROM Version: v3.0

[Normal mode]
BootFromNORFlash
[SPIF Err]Invalid ID
[Start Boot ROM...]
=== Load PARTBL ===
=== Load Done ===
=== Load ISP_IQ ===
[fcs chk pass]
ISP_IQ @ 0x8401080, 0x19f80, 0x0
mfcs_data version 0x00010001
fcs_data version 0x00010101
=== Process ISP_IQ ===
=== Load Done ===
=== Load BL ===
[Image Start Table @ 0x18200]
=== Load Done ===
```

Fig 6-2 boot up console log

6.5 Image Download

6.5.1 Enter Download Mode

Press these two buttons in following figure simultaneously to enter download mode.

Fig 6-3 Enter download mode

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

6.5.2 Download Image to Flash

Copy the image **flash_ntz.bin** to image tool folder **tools/Pro2_PG_tool_linux_v1.3.x**.
Then, Use Pro2_PG_tool_linux_v1.3.x command line tool to download image.

Nor flash

$ ./uartfwburn.linux -p /dev/ttyUSB? -f flash_ntz.bin -b 2000000 -U

Nand flash

$ ./uartfwburn.linux -p /dev/ttyUSB? -f flash_ntz.bin -b 2000000 -n pro2

After firmware image downloaded, press the reset button to reboot the device and open terminal console to check the log.

Note: If using windows, replace *uartfwburn.linux* with *uartfwburn.exe* and replace */dev/ttyUSB?* with *COM?*

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

7 MQTT Demo

7.1 Run MQTT Demo

Default setting of SDK are enable MQTT demo. Please refer 5.4.2 to enable the MQTT demo and refer chapter 6 to build and download the demo image.
Once the AmebaPro2 EVB has rebooted, the application will automatically start run MQTT demo and communicate to IoT Core.

```
[Driver]: set ssid [RealEZ]
                          [RF] [RFK] Tx pause!!
[RF] [RFK] Tx pause!!

[Driver]: start auth to [redacted]

[Driver]: auth alg = 2

[Driver]: auth success, start assoc

[Driver]: association success(res=28)

[Driver]: wlan0: DL RSVD page success! DLBcnCount:1, poll:1
                                                    0 301 [example_ama] Write certificate...

1 408 [iot_thread] [INFO ][DEMO][408] ---------STARTING DEMO---------


2 414 [iot_thread] [INFO ][INIT][414] SDK successfully initialized.
```

…

```
Interface 0 IP address : 192.168.[redacted]
                                  3 53555 [iot_thread] [INFO ][DEMO][53555] Successfully initialized the demo. N
etwork type for the demo: 1

4 53564 [iot_thread] [INFO] Creating a TLS connection to [redacted]-ats.iot.ap-southeast-1.amazonaws.com:8883.
5 54778 [iot_thread] [INFO] Creating an MQTT connection to [redacted]-ats.iot.ap-southeast-1.amazonaws.com.
6 54909 [iot_thread] [INFO] Packet received. ReceivedBytes=2.
7 54913 [iot_thread] [INFO] CONNACK session present bit not set.
8 54919 [iot_thread] [INFO] Connection accepted.
9 54924 [iot_thread] [INFO] Received MQTT CONNACK successfully from broker.
10 54930 [iot_thread] [INFO] MQTT connection established with the broker.
11 54937 [iot_thread] [INFO] An MQTT connection is established with [redacted]-ats.iot.ap-southeast-1.amazonaws.c
om.
12 54949 [iot_thread] [INFO] Attempt to subscribe to the MQTT topic ameba-ota/example/topic.
13 54956 [iot_thread] [INFO] SUBSCRIBE sent for topic ameba-ota/example/topic to broker.
14 55070 [iot_thread] [INFO] Packet received. ReceivedBytes=3.
15 55074 [iot_thread] [INFO] Subscribed to the topic ameba-ota/example/topic with maximum QoS 1.
16 56082 [iot_thread] [INFO] Publish to the MQTT topic ameba-ota/example/topic.
17 56087 [iot_thread] [INFO] Attempt to receive publish message from broker.
18 56241 [iot_thread] [INFO] Packet received. ReceivedBytes=2.
19 56246 [iot_thread] [INFO] Ack packet deserialized with result: MQTTSuccess.
20 56252 [iot_thread] [INFO] State record updated. New state=MQTTPublishDone.
21 56259 [iot_thread] [INFO] PUBACK received for packet Id 2.
22 56265 [iot_thread] [INFO] Packet received. ReceivedBytes=39.
23 56270 [iot_thread] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
24 56280 [iot_thread] [INFO] State record updated. New state=MQTTPubAckSend.
25 56286 [iot_thread] [INFO] Incoming QoS : 1
```

…

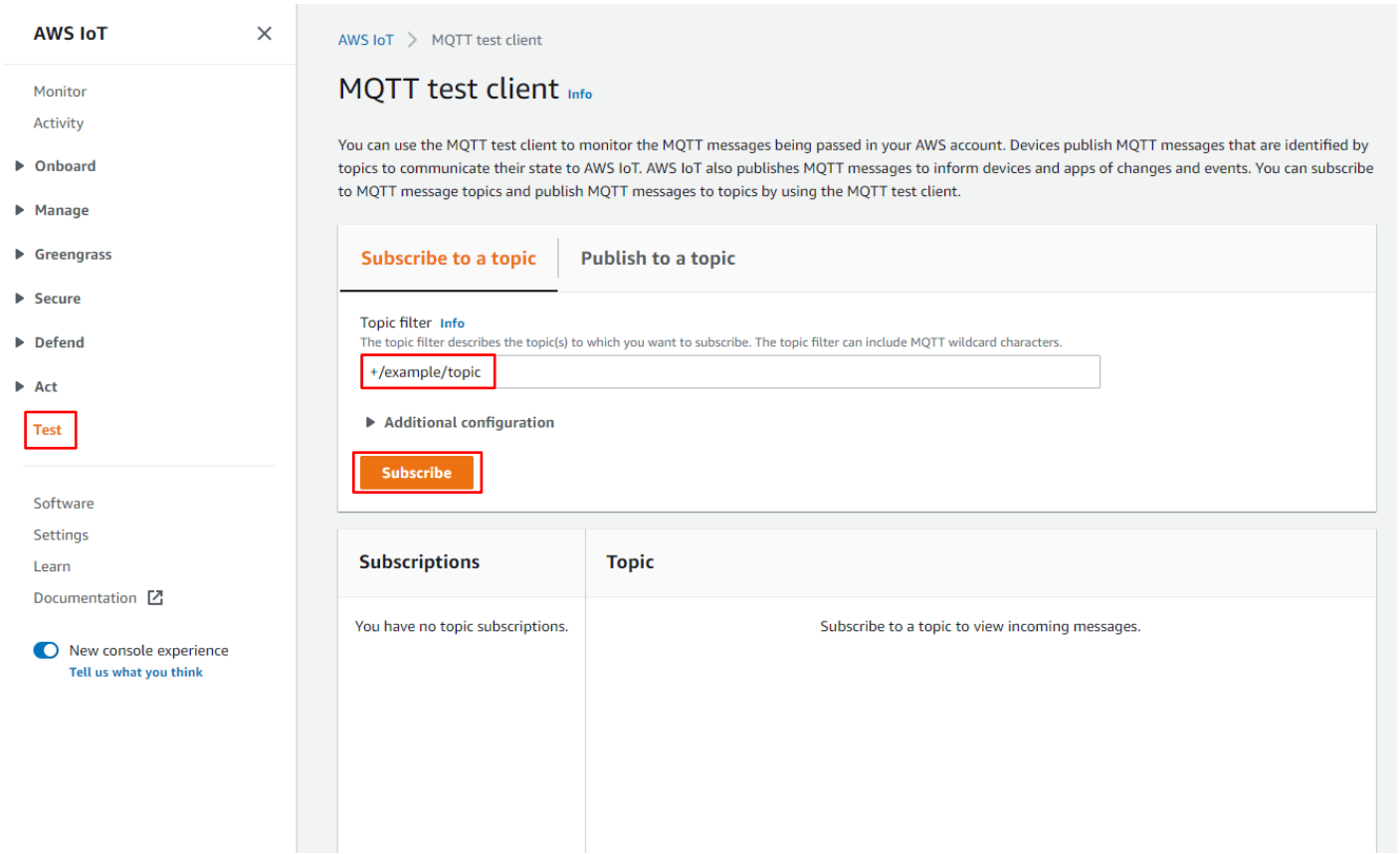
All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

```
248 122674 [iot_thread] [INFO] Demo run is successful with 3 successful loops out of total 3 loops.
249 123681 [iot_thread] [INFO ][DEMO][123681] Demo completed successfully.


Deinitializing WIFI ...
WIFI deinitialized250 123809 [iot_thread] [INFO ][INIT][123809] SDK cleanup done.

251 123813 [iot_thread] [INFO ][DEMO][123813] -------DEMO FINISHED-------
```

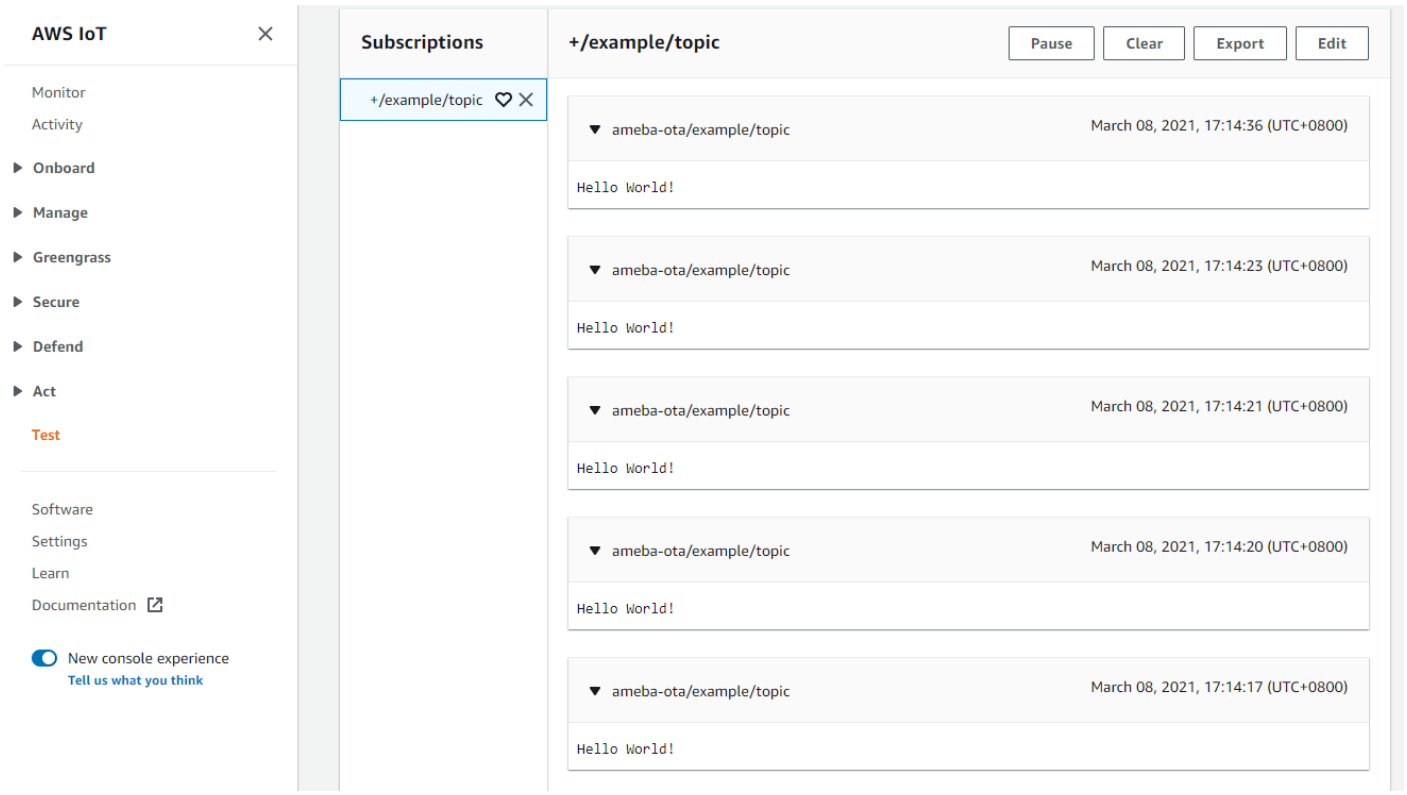
7.2 Monitoring MQTT Messages on the Cloud

To subscribe to the MQTT topic with the AWS IoT MQTT client

1. Sign in to the AWS IoT console.
2. In the navigation pane, choose Test to open the MQTT client.
3. In Subscription topic, enter "+/example/topic", and then choose Subscribe to topic.



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.



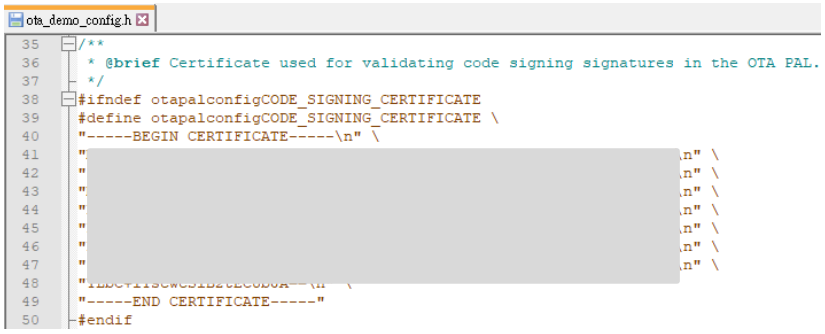
All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

8 OTA Demo

8.1 OTA Update Prerequisites

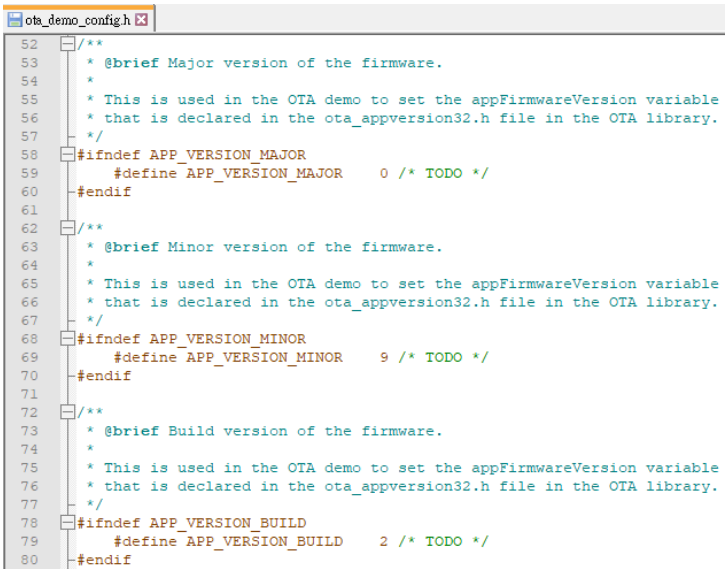
Please refer to the AWS official guide (https://docs.aws.amazon.com/freertos/latest/userguide/ota-prereqs.html) and finish the following steps:

Step 1. Prerequisites for OTA updates using MQTT
Step 2. Create an Amazon S3 bucket to store your update
Step 3. Create an OTA Update service role
Step 4. Create an OTA user policy
Step 5. Create esdsasigner.key and ecdsasigner.crt by openSSL
you can create the key and certification by running:
$ sudo openssl ecparam -name prime256v1 -genkey -out ecdsa-sha256-signer.key.pem
$ sudo openssl req -new -x509 -days 3650 -key ecdsa-sha256-signer.key.pem -out ecdsa-sha256-signer.crt.pem
Step 6. Add certificate pem(ecdsa-sha256-signer.crt.pem) into *project\realtek_amebapro2_v0_example\src\aws_iot_freertos_lts\vendors\realtek\boards\amebaPro2\aws_demos\config_files\ota_demo_config.h*



8.2 Set the App Version to Image File

The app number in “ota_demo_config.h” decide the version of application code:



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

Please note that the newer image file must have the bigger version number. So now, you need two image file to perform this demo.

- One image with older version should be downloaded to your AmebaPro2, and wait the OTA job coming.
- Another image with newer version (APP_VERSION_MAJOR or APP_VERSION_MINOR should be incremented and the image built) will be uploaded to S3 bucket. Then, create a new job for OTA.

Your newer image for OTA - **ota.bin** will also be located in "build/" directory after compilation.

Note: newer version image file should be signed by a private key before uploading. Next section will introduce how to sign the image.

8.3 Custom-Signed Image and Signature

We use custom signing feature provided by amazon to manually sign the OTA binary to get the signatures used to verify the integrity of the firmware after download. The ota.bin is manually signed using the ECDSA P-256 key provided by user.

The custom signing process is executed by a python script – **python_custom_ecdsa_Pro2.py**, that provided in the folder "project\realtek_amebapro2_v0_example\src\aws_iot_freertos_lts\tools\amazon_ota_tools\python_custom_ecdsa_Pro2.py"

The python script requires the following pre-requisites to work

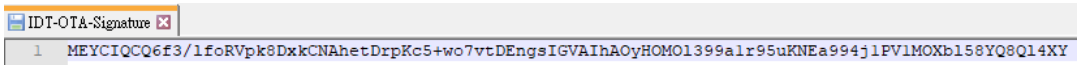
1. Python must be installed in the windows system with version 3.7.x or later
2. Pyopenssl library must be installed using 'pip install pyopenssl'
3. The ECDSA signing key and the Certificate pair must be present in the same folder as the python script and must be named 'ecdsa-sha256-signer.key.pem' and 'ecdsa-sha256-signer.crt.pem' respectively.
   (**Note:** *The key pair in SDK are just for example, please generated new key by openssl !*)

Run the python script in folder: "project\realtek_amebapro2_v0_example\src\aws_iot_freertos_lts\tools\amazon_ota_tools\"

- command after GCC build : **$ python3 python_custom_ecdsa_Pro2.py**

There might be some error if there are packages lack in your environment (like openssl...). Please install the package and run the script again.

Once all these are present and the python script is run, it will generate a signature file – **IDT-OTA-Signature** in "project\realtek_amebapro2_v0_example\src\aws_iot_freertos_lts\tools\amazon_ota_tools". This will be used in next step – create OTA job in AWS-IOT Core.



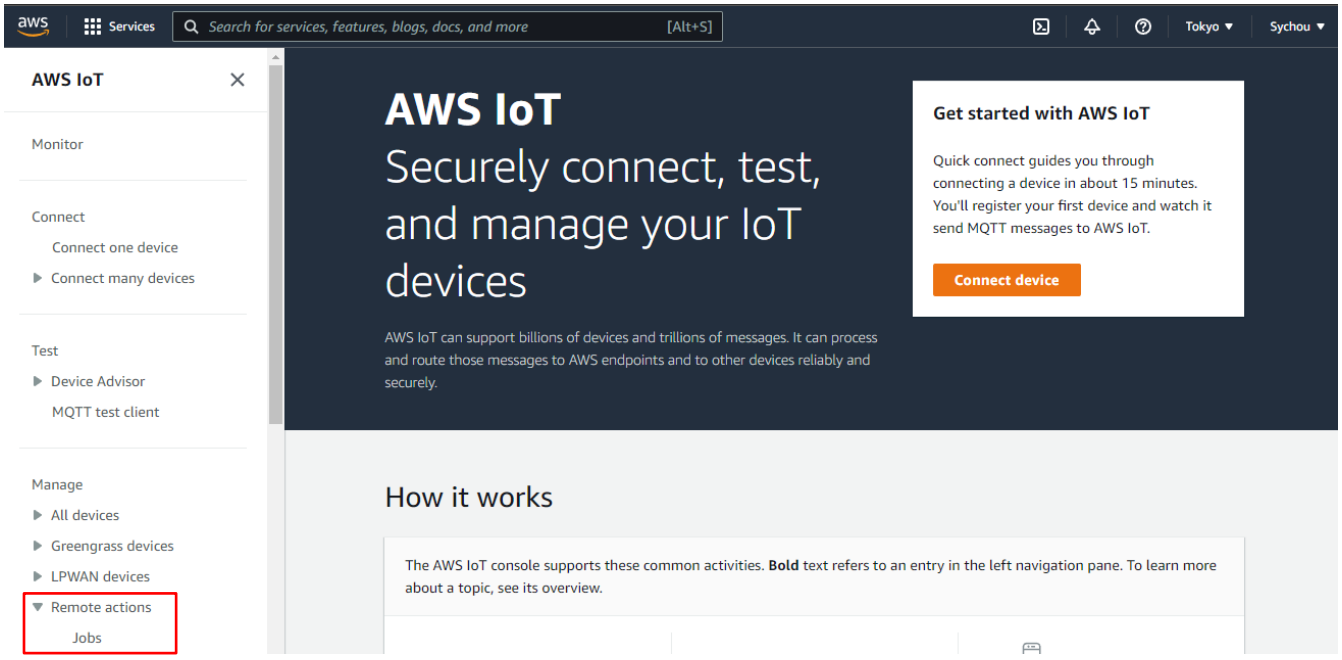
After getting the custom signed signature, you can upload **ota.bin** to your S3 bucket.

8.4 How to Trigger a Custom Signed OTA Job in Amazon AWS IOT Core

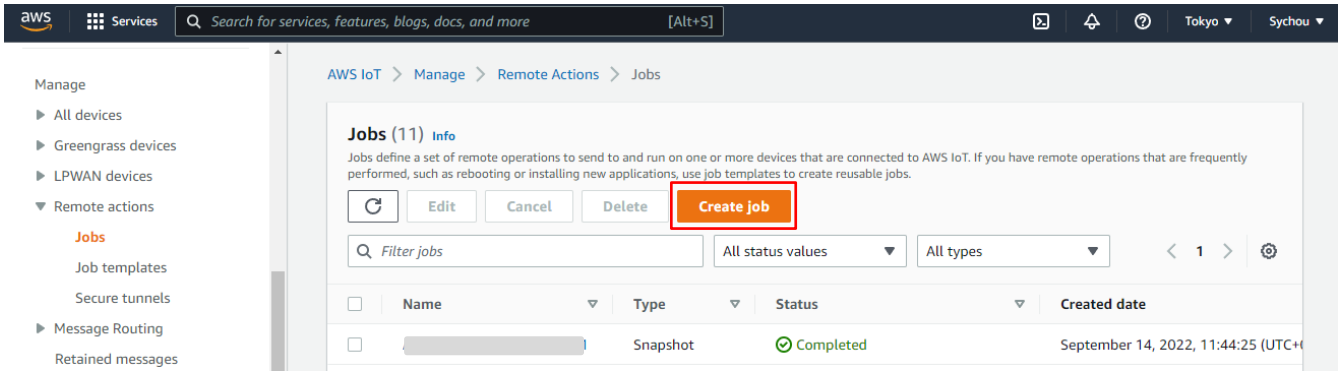
Go to AWS IoT Core https://console.aws.amazon.com/iot/home. Then, follow the following steps to create an AWS OTA task for AmebaPro2:

Step 1. Click on 'Create OTA update job', select your job type and then click next.

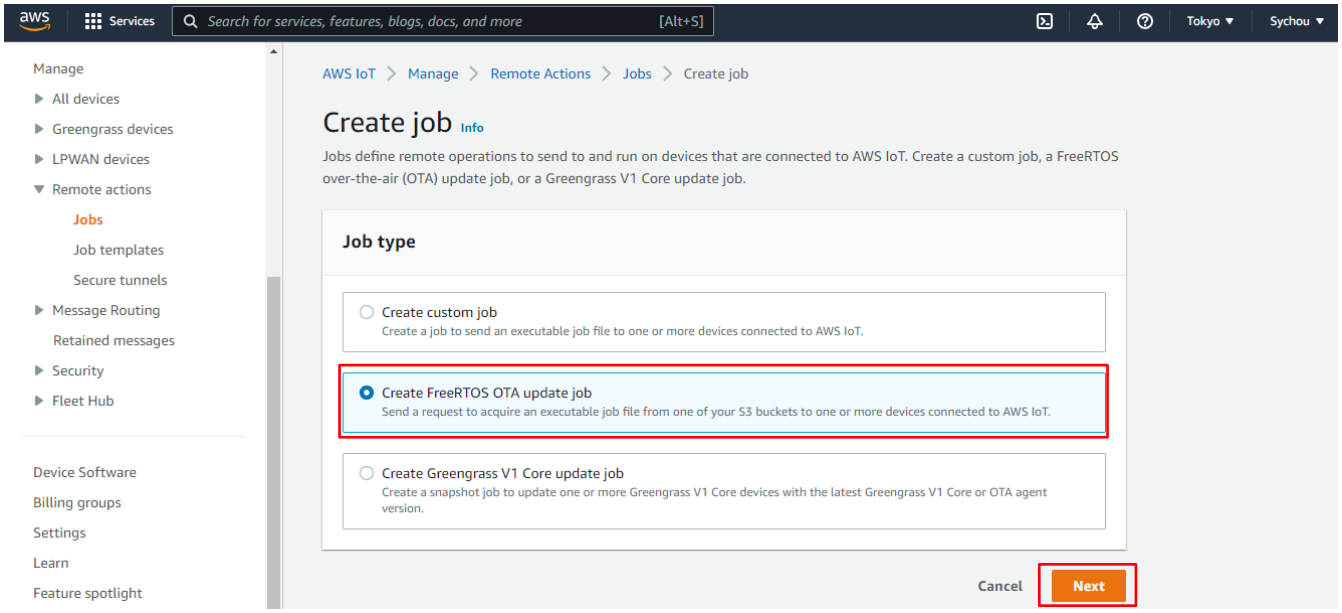
All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.



...

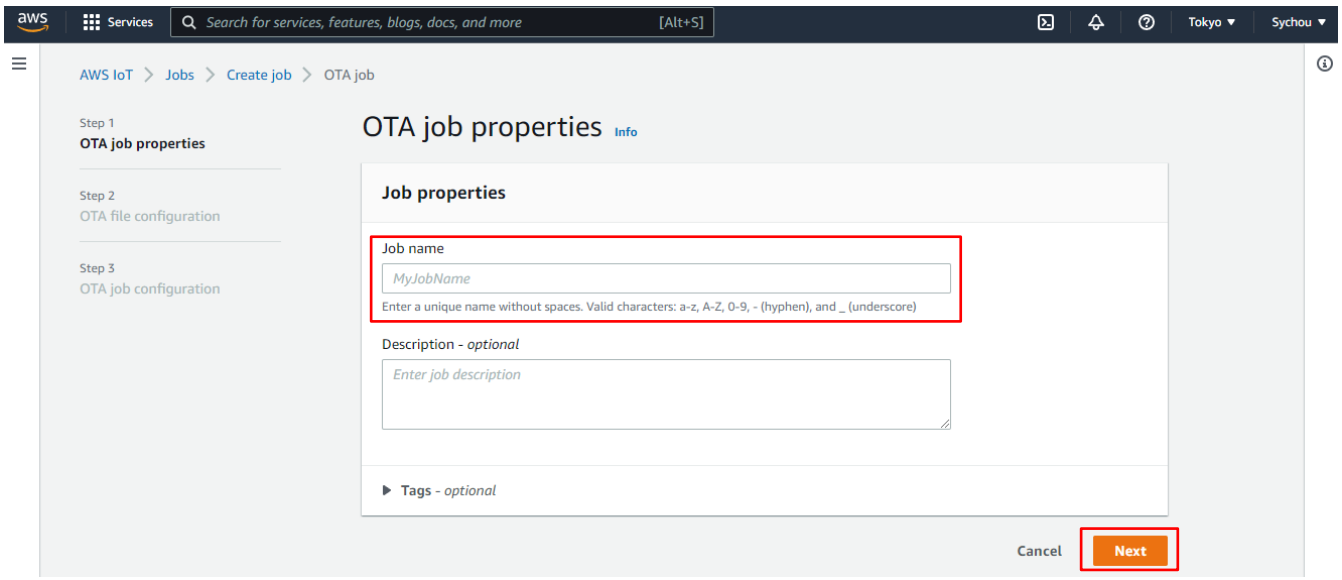


...

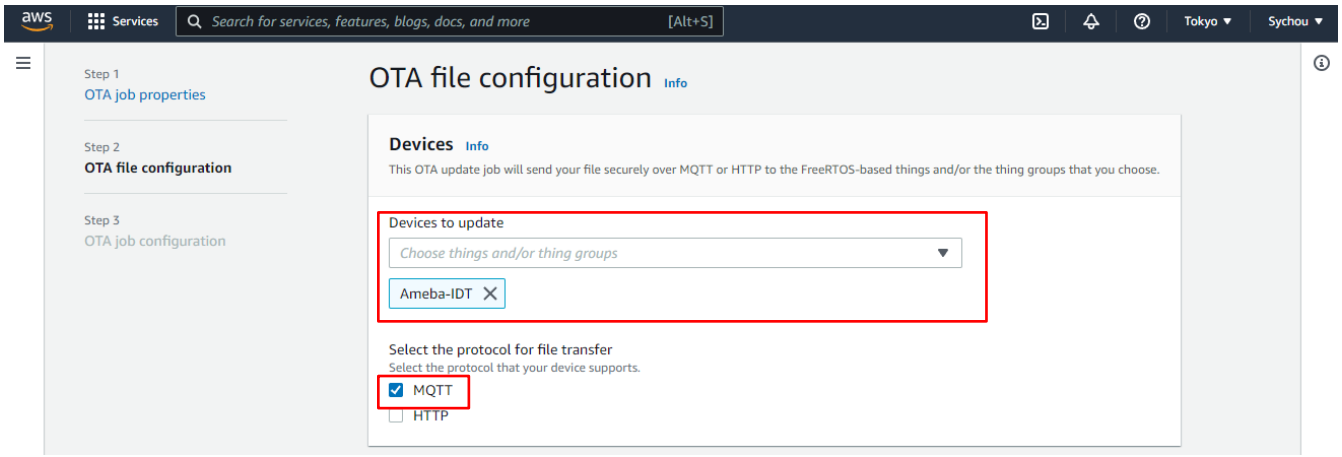


All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

Step 2. For Job properties, give a unique name to your OTA job, then click next.

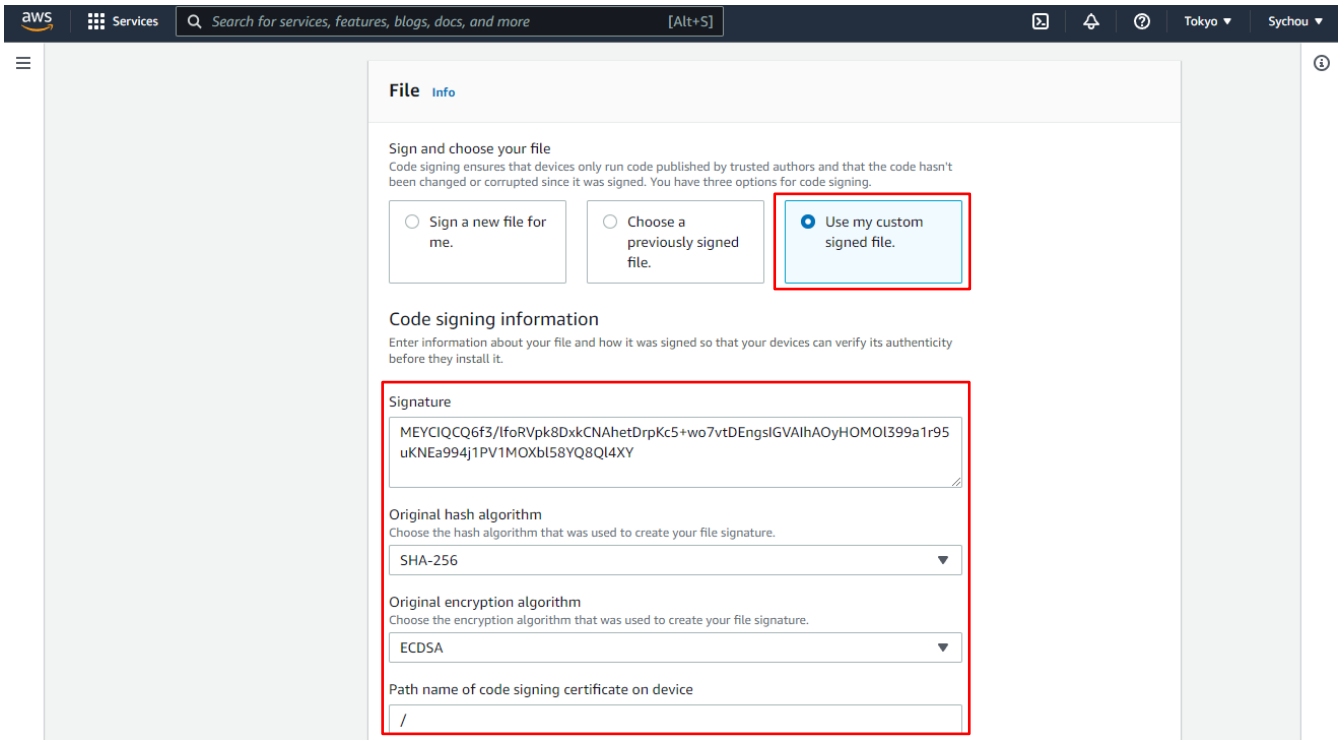


Step 3. In the following page, choose your device to update and select the protocol for file transfer

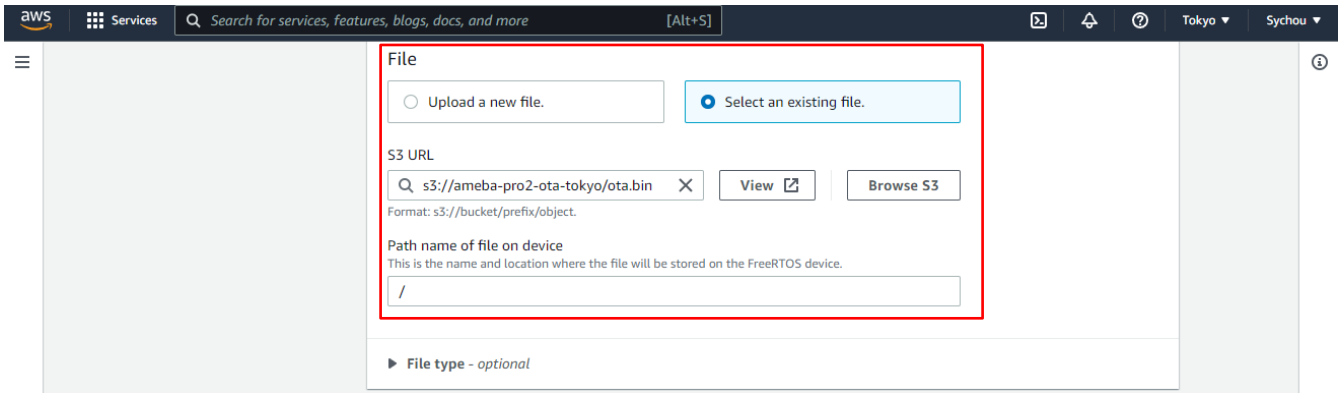


All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

Step 4. In the following page, choose the option 'Use my custom signed firmware image'.
In the signature field, copy and paste the content of **IDT-OTA-Signature** (generated in previous section).
Choose hash algorithm as 'SHA-256'.
Choose encryption algorithm as 'ECDSA'.
In "pathname of code signing certificate", enter '/'

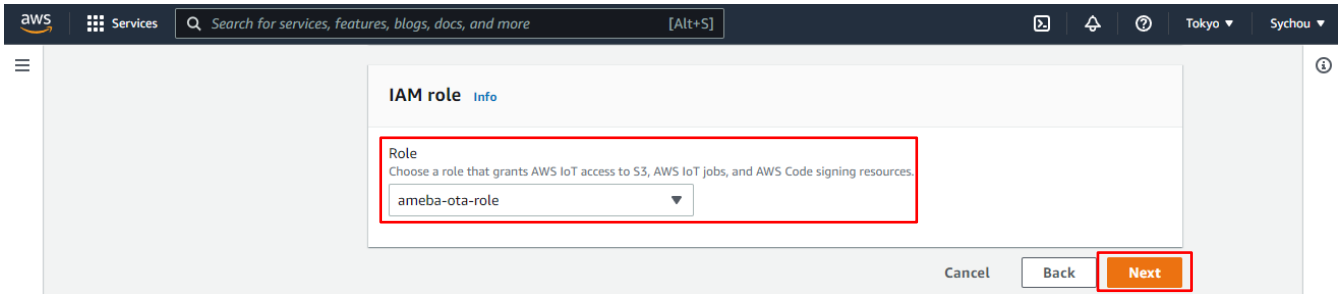


Step 5. Choose your custom signed firmware binary that was generated by the python script from S3 bucket.
In "Pathname of file on device", enter '/'

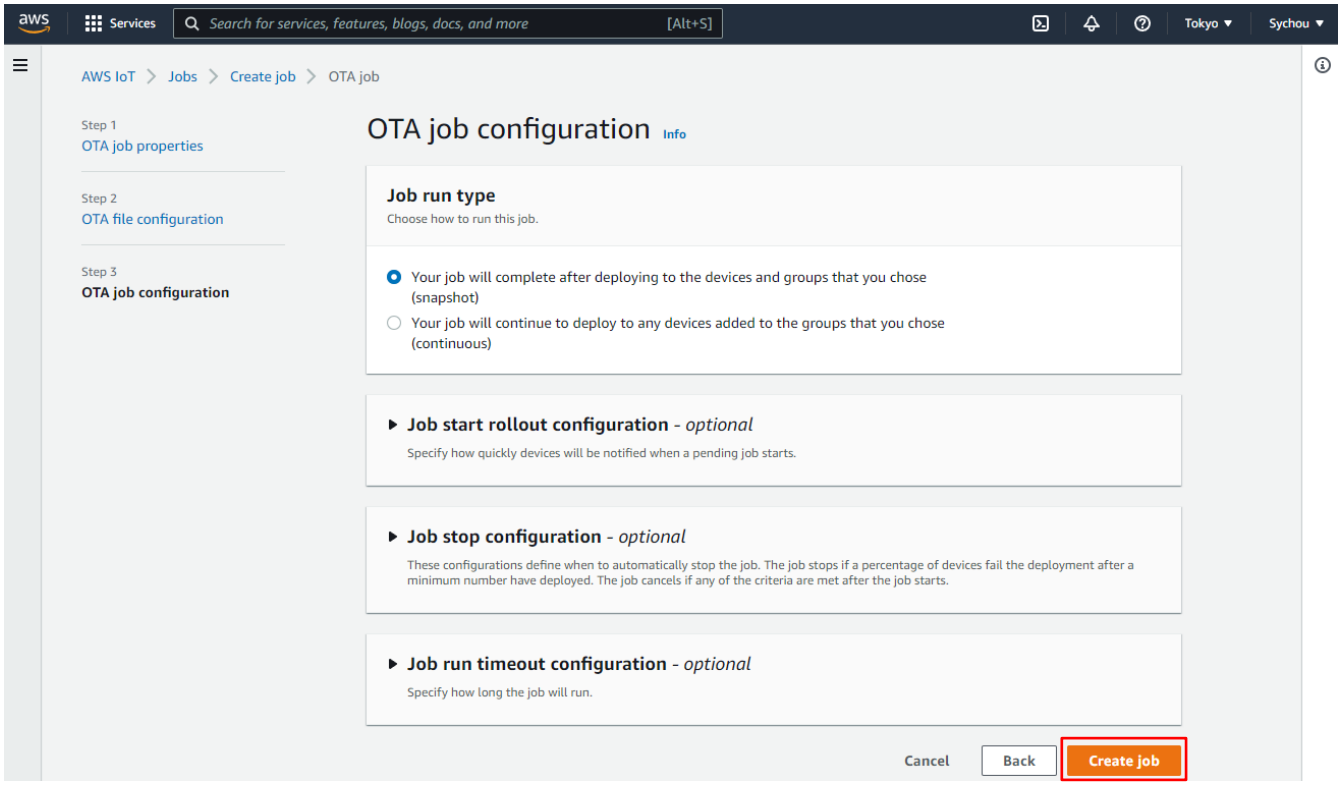


All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

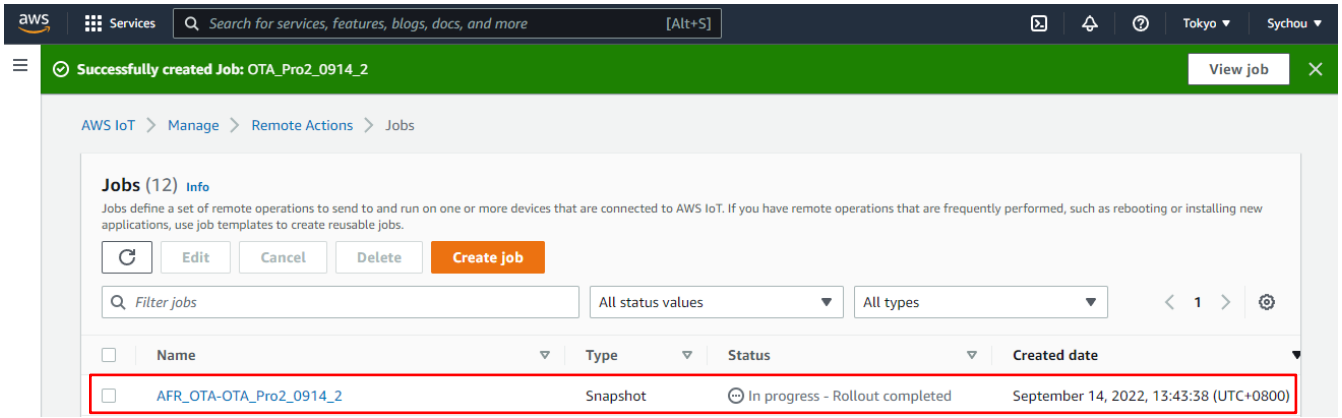
Step 7. Choose the IAM role for OTA update job. User can create a role according to the instructions in Create an OTA Update service role.



Step 8. Click next, and create your OTA job.



...



All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

8.5 Run OTA Demo

Now we can see that the status of OTA job on AWS IoT Core is “in progress”. It means that it is waiting AmebaPro2 to request the update.

Next, download the image file with older version number 0.9.2 to AmebaPro2 and then reboot the device, the application will automatically start run OTA demo.

In the beginning, we can check the app version of this running firmware, and the OTA process by the job ID:

```
3 6150 [iot_thread] [INFO ][DEMO][6150] Successfully initialized the demo. Network type for the demo: 1

4 6159 [iot_thread] [INFO] OTA over MQTT demo, Application version 0.9.2
5 6166 [iot_thread] [INFO] Creating a TLS connection to a2zweh2b7yb784-ats.iot.ap-northeast-1.amazonaws.com:8883.
6 7233 [iot_thread] [INFO] Creating an MQTT connection to a2zweh2b7yb784-ats.iot.ap-northeast-1.amazonaws.com.
7 7372 [iot_thread] [INFO] Packet received. ReceivedBytes=2.
8 7376 [iot_thread] [INFO] CONNACK session present bit not set.
9 7381 [iot_thread] [INFO] Connection accepted.
10 7386 [iot_thread] [INFO] Received MQTT CONNACK successfully from broker.
11 7393 [iot_thread] [INFO] MQTT connection established with the broker.
12 7399 [iot_thread] [INFO]  Received: 0   Queued: 0   Processed: 0   Dropped: 0
13 7406 [OTA Agent T] [INFO] otaPal_GetPlatformImageState
14 7413 [OTA Agent T] [INFO] [prvPAL_GetPlatformImageState_amebaPro2] Image current state (0x02).
15 7420 [OTA Agent T] [INFO] Current State=[RequestingJob], Event=[Start], New state=[RequestingJob]
16 7564 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=3.
17 7568 [OTA Agent T] [INFO] SUBSCRIBED to topic $aws/things/Ameba-IDT/jobs/notify-next to broker.


18 7577 [OTA Agent T] [INFO] Subscribed to MQTT topic: $aws/things/Ameba-IDT/jobs/notify-next
19 8068 [MQTT Agent ] [INFO] Publishing message to $aws/things/Ameba-IDT/jobs/$next/get.

20 8186 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=2.
21 8190 [MQTT Agent ] [INFO] Ack packet deserialized with result: MQTTSuccess.
22 8197 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
23 8205 [OTA Agent T] [INFO] Sent PUBLISH packet to broker $aws/things/Ameba-IDT/jobs/$next/get to broker.


24 8214 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=617.
25 8220 [OTA Agent T] [WARN] OTA Timer handle NULL for Timerid=0, can't stop.
26 8227 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
28 8236 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
27 8235 [OTA Agent T] [INFO] Current State=[WaitingForJob], Event=[RequestJobDocument], New state=[WaitingForJob]
29 8252 [MQTT Agent ] [INFO] Received job message callback, size 570.


30 8259 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobId: AFR_OTA-OTA_Pro2_0914_1]
31 8269 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobDocument.afr_ota.streamname: AFR_OTA-cde03295-4b6
6-46e5-8564-6e6075f395c6]
32 8282 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[execution.jobDocument.afr_ota.protocols: ["MQTT"]]
33 8292 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[filepath: /]
34 8299 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[filesize: 2248708]
35 8306 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[fileid: 0]
36 8313 [OTA Agent T] [INFO] Extracted parameter: [key: value]=[certfile: /]
37 8320 [OTA Agent T] [INFO] Extracted parameter: [ sig-sha256-ecdsa: MEYCIQCWDIKQJvcPrNgw0Co/mRmBcIoR... ]
38 8329 [OTA Agent T] [INFO] Job document was accepted. Attempting to begin the update.
39 8337 [OTA Agent T] [INFO] Job parsing success: OtaJobParseErr_t=OtaJobParseErrNone, Job name=AFR_OTA-OTA_Pro2_0914_1
```


All information provided in this document is subject to legal disclaimers.

© REALTEK 2022. All rights reserved.

We can see that the OTA process start!

```
41 8354 [OTA Agent T] [INFO] [prvPAL_GetPlatformImageState_amebaPro2] Image current state (0x02).
42 8362 [OTA Agent T] [INFO] otaPal_CreateFileForRx
43 8366 [OTA Agent T] [INFO] [prvPAL_CreateFileForRx_amebaPro2] Current firmware index is 1
44 8375 [OTA Agent T] [INFO] [prvPAL_CreateFileForRx_amebaPro2] target_fw_addr=0x4c0000, target_fw_len=0x300000
45 8385 [OTA Agent T] [INFO] otaPal_GetPlatformImageState
46 8391 [OTA Agent T] [INFO] [prvPAL_GetPlatformImageState_amebaPro2] Image current state (0x02).
47 8399 [OTA Agent T] [INFO] Setting OTA data interface.
48 8404 [OTA Agent T] [INFO] Current State=[CreatingFile], Event=[ReceivedJobDocument], New state=[CreatingFile]
49 8414 [iot_thread] [INFO]  Received: 0   Queued: 0   Processed: 0   Dropped: 0
50 8878 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=3.
51 8882 [OTA Agent T] [INFO] SUBSCRIBED to topic $aws/things/Ameba-IDT/streams/AFR_OTA-cde03295-4b66-46e5-8564-6e6075f395c6/da
ta/cbor to broker.


52 8896 [OTA Agent T] [INFO] Current State=[RequestingFileBlock], Event=[CreateFile], New state=[RequestingFileBlock]
53 9382 [MQTT Agent ] [INFO] Publishing message to $aws/things/Ameba-IDT/streams/AFR_OTA-cde03295-4b66-46e5-8564-6e6075f395c6/
get/cbor.

54 9393 [OTA Agent T] [INFO] Sent PUBLISH packet to broker $aws/things/Ameba-IDT/streams/AFR_OTA-cde03295-4b66-46e5-8564-6e607
5f395c6/get/cbor to broker.


55 9406 [OTA Agent T] [INFO] Published to MQTT topic to request the next block: topic=$aws/things/Ameba-IDT/streams/AFR_OTA-cd
e03295-4b66-46e5-8564-6e6075f395c6/get/cbor
56 9422 [iot_thread] [INFO]  Received: 0   Queued: 0   Processed: 0   Dropped: 0
57 9428 [OTA Agent T] [INFO] Current State=[WaitingForFileBlock], Event=[RequestFileBlock], New state=[WaitingForFileBlock]
58 9541 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=4206.
59 9546 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
60 9555 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
61 9562 [MQTT Agent ] [INFO] Received data message callback, size 4120.


62 9569 [OTA Agent T] [INFO] Received valid file block: Block index=0, Size=4096
63 9577 [OTA Agent T] [INFO] otaPal_WriteBlock
64 9582 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] C->fileSize 2248708, iOffset: 0x0: iBlockSize: 0x1000
65 9594 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] ota_len:2248708, cur_block:0
66 9602 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] FIRST image data arrived 4096, back up the first 8-bytes fw label
67 9612 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] label backup get [8084971200083107968]
68 9629 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=4206.
69 9635 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
70 9645 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
71 9651 [MQTT Agent ] [INFO] Received data message callback, size 4120.


72 9658 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=4206.
73 9664 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
74 9673 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
76 9680 [MQTT Agent ] [INFO] Received data message callback, size 4120.


75 9679 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] Write bytes: read_bytes 4096, ulBlockSize 4096
77 9696 [OTA Agent T] [INFO] Number of blocks remaining: 549
78 9700 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=4206.
79 9706 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
80 9715 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
81 9722 [MQTT Agent ] [INFO] Received data message callback, size 4120.
```

After receiving the final block, the signature will be checked if valid or not. If signature is valid, the OTA process is successful!
Then, the device will reboot with new firmware automatically.

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

```
6441 100915 [OTA Agent T] [INFO] Received valid file block: Block index=549, Size=4
6442 100922 [OTA Agent T] [INFO] otaPal_WriteBlock
6443 100927 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] C->fileSize 2248708, iOffset: 0x225000: iBlockSize: 0x4
6444 100938 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] ota_len:2248708, cur_block:549
6445 100946 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] LAST image data arrived
6446 100964 [iot_thread] [INFO]  Received: 550   Queued: 550   Processed: 549   Dropped: 0
6447 101004 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] Write bytes: read_bytes 4, ulBlockSize 4
6448 101011 [OTA Agent T] [INFO] Received final block of the update.
6449 101018 [OTA Agent T] [INFO] otaPal_CloseFile
6450 101022 [OTA Agent T] [INFO] [prvPAL_CloseFile_amebaPro2] Authenticating and closing file.
6451 101030 [OTA Agent T] [INFO] [prvPAL_CheckFileSignature_amebaPro2] Started sig-sha256-ecdsa signature verification, file:
/
6452 101042 [OTA Agent T] [INFO] [prvPAL_ReadAndAssumeCertificate_amebaPro2] Assume Cert - No such file: /. Using header file
6453 101053 [OTA Agent T] [INFO] [prvSignatureVerificationUpdate_amebaPro2] fw2 address will be upgraded
6454 101392 [OTA Agent T] [INFO] [prvSignatureVerificationUpdate_amebaPro2] Signature Verification Update done.
6455 101565 [OTA Agent T] [INFO] [prvPAL_CloseFile_amebaPro2] sig-sha256-ecdsa signature verification passed.
6456 101573 [OTA Agent T] [INFO] Received entire update and validated the signature.
6457 101580 [MQTT Agent ] [INFO] Publishing message to $aws/things/Ameba-IDT/jobs/AFR_OTA-OTA_Pro2_0914_1/update.

6458 101745 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=2.
6459 101750 [MQTT Agent ] [INFO] Ack packet deserialized with result: MQTTSuccess.
6460 101757 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
6461 101765 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=92.
6462 101771 [OTA Agent T] [INFO] Sent PUBLISH packet to broker $aws/things/Ameba-IDT/jobs/AFR_OTA-OTA_Pro2_0914_1/update to br
oker.


6463 101783 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
6464 101792 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
6465 101799 [OTA Agent T] [INFO] Received OtaJobEventActivate callback from OTA Agent.
6466 101807 [OTA Agent T] [INFO] otaPal_ActivateNewImage
6467 101812 [OTA Agent T] [INFO] [prvPAL_ActivateNewImage_amebaPro2] fw2 need to be activated!
6468 101820 [OTA Agent T] [INFO] [prvPAL_ActivateNewImage_amebaPro2] Append FW label
6469 101828 [OTA Agent T] [INFO] [prvPAL_ActivateNewImage_amebaPro2] FW label:
 52 54 4C 38 37 33 35 42
6470 101837 [OTA Agent T] [INFO] prvPAL_SetPlatformImageState_amebaPro2
6471 101867 [OTA Agent T] [INFO] [prvPAL_ActivateNewImage_amebaPro2] Resetting MCU to activate new image.
```

After booting with newer image, the device will start a self-test mode to check the app version is newer than before.
We can see that the version now is 0.9.3, which is newer than old one 0.9.2.

```
38 8023 [OTA Agent T] [INFO] Extracted parameter [ sig-sha256-ecdsa: MEYCIQCWDIKQJvcPrNqw0Co/mRmBcIoR... ]
39 8032 [OTA Agent T] [INFO] In self test mode.
40 8036 [OTA Agent T] [INFO] New image has a higher version number than the current image: New image version=0.9.3, Previous i
mage version=0.9.2
41 8049 [OTA Agent T] [INFO] Image version is valid: Begin testing file: File ID=0
42 8057 [OTA Agent T] [INFO] otaPal_SetPlatformImageState
43 8062 [OTA Agent T] [INFO] prvPAL_SetPlatformImageState_amebaPro2
44 8115 [iot_thread] [INFO]  Received: 0   Queued: 0   Processed: 0   Dropped: 0
45 8453 [MQTT Agent ] [INFO] Publishing message to $aws/things/Ameba-IDT/jobs/AFR_OTA-OTA_Pro2_0914_1/update.

46 8616 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=2.
47 8621 [MQTT Agent ] [INFO] Ack packet deserialized with result: MQTTSuccess.
48 8628 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
49 8635 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=92.
50 8641 [OTA Agent T] [INFO] Sent PUBLISH packet to broker $aws/things/Ameba-IDT/jobs/AFR_OTA-OTA_Pro2_0914_1/update to broker
.


51 8652 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
52 8661 [OTA Agent T] [INFO] Job parsing success: OtaJobParseErr_t=OtaJobParseErrNone, Job name=AFR_OTA-OTA_Pro2_0914_1
53 8672 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
54 8678 [OTA Agent T] [INFO] otaPal_GetPlatformImageState
55 8685 [OTA Agent T] [INFO] [prvPAL_GetPlatformImageState_amebaPro2] Image current state (0x01).
56 8693 [OTA Agent T] [INFO] Current State=[CreatingFile], Event=[ReceivedJobDocument], New state=[CreatingFile]
57 8703 [OTA Agent T] [INFO] Beginning self-test.
58 8707 [OTA Agent T] [INFO] otaPal_GetPlatformImageState
59 8714 [OTA Agent T] [INFO] [prvPAL_GetPlatformImageState_amebaPro2] Image current state (0x01).
60 8721 [OTA Agent T] [INFO] Received OtaJobEventStartTest callback from OTA Agent.
61 8729 [OTA Agent T] [INFO] otaPal_SetPlatformImageState
62 8734 [OTA Agent T] [INFO] prvPAL_SetPlatformImageState_amebaPro2
63 8742 [OTA Agent T] [INFO] [prvPAL_GetPlatformImageState_amebaPro2] Image current state (0x01).
64 9120 [iot_thread] [INFO]  Received: 0   Queued: 0   Processed: 0   Dropped: 0
65 9184 [MQTT Agent ] [INFO] Publishing message to $aws/things/Ameba-IDT/jobs/AFR_OTA-OTA_Pro2_0914_1/update.

66 9350 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=2.
67 9354 [MQTT Agent ] [INFO] Ack packet deserialized with result: MQTTSuccess.
68 9361 [MQTT Agent ] [INFO] State record updated. New state=MQTTPublishDone.
69 9369 [OTA Agent T] [INFO] Sent PUBLISH packet to broker $aws/things/Ameba-IDT/jobs/AFR_OTA-OTA_Pro2_0914_1/update to broker
.


70 9380 [MQTT Agent ] [INFO] Packet received. ReceivedBytes=92.
71 9386 [OTA Agent T] [INFO] Successfully updated with the new image.
72 9392 [MQTT Agent ] [INFO] De-serialized incoming PUBLISH packet: DeserializerResult=MQTTSuccess.
```

In the final, the log imply that OTA success!

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

9 Troubleshooting

If these steps don't work, look at the device log in the serial terminal. You should see some text that indicates the source of the problem.

For general troubleshooting information about Getting Started with FreeRTOS, see Troubleshooting getting started.

9.1 ERROR: Invalid Key

Please check **WIFI_SSID** and **WIFI_PASSWORD** in in "project/realtek_amebapro2_v0_example/src/aws_iot_freertos_lts/demos/include /aws_clientcredential.h"

```
Enter SSID for Soft AP started
3 1098 [example_a] Wi-Fi configuration successful.
4 1108 [iot_threa] [INFO ][DEMO][1108] ---------STARTING DEMO---------

5 1115 [iot_threa] [INFO ][INIT][1115] SDK successfully initialized.

LwIP_DHCP: dhcp stop.
Deinitializing WIFI ...
WIFI deinitialized
Initializing WIFI ...
WIFI initialized

Joining BSS by SSID ...

ERROR:Invalid Key
ERROR: Can't connect to AP
Joining BSS by SSID ...

ERROR:Invalid Key
ERROR: Can't connect to AP
Joining BSS by SSID ...
```

9.2 Failed to Establish New MQTT Connection

Please check **clientcredentialMQTT_BROKER_ENDPOINT** in "project/realtek_amebapro2_v0_example/src/aws_iot_freertos_lts/demos/include/aws_clientcredential.h"

```
6 12508 [iot_threa] [INFO ][DEMO][12508] Successfully initialized the demo. Network type for the demo: 1
7 12517 [iot_threa] [INFO ][MQTT][12517] MQTT library successfully initialized.
8 12524 [iot_threa] [INFO ][DEMO][12524] MQTT demo client identifier is ameba-ota (length 9).
9 12624 [iot_threa] [ERROR][NET][12624] Failed to resolve [redacted].amazonaws.com.
10 12934 [iot_threa] [ERROR][MQTT][12934] Failed to establish new MQTT connection, error NETWORK ERROR.
11 12943 [iot_threa] [ERROR][DEMO][12943] MQTT CONNECT returned error NETWORK ERROR.
12 12951 [iot_threa] [INFO ][MQTT][12950] MQTT library cleanup done.
13 12957 [iot_threa] [ERROR][DEMO][12957] Error running demo.
Interface 0 IP address : 192.168.90.185
LwIP_DHCP: dhcp stop.
Deinitializing WIFI ...
14 13094 [iot_threa] [INFO ][INIT][13094] SDK cleanup done.
15 13099 [iot_threa] [INFO ][DEMO][13099] -------DEMO FINISHED-------
```

9.3 TLS_Connect Fail

Please check **keyCLIENT_CERTIFICATE_PEM** and **keyCLIENT_PRIVATE_KEY_PEM** in "project/realtek_amebapro2_v0_example/src/aws_iot_freertos_lts/demos/include/aws_clientcredential_keys.h"

```
8 13501 [iot_threa] [INFO ][DEMO][13501] Successfully initialized the demo. Network type for the demo: 1
9 13511 [iot_threa] [INFO ][MQTT][13511] MQTT library successfully initialized.
10 13518 [iot_threa] [INFO ][DEMO][13518] MQTT demo client identifier is ameba-ota (length 9).
11 20102 [iot_threa] ERROR: Private key not found. 12 20107 [iot_threa] TLS Connect fail (0x7d4, [redacted].amazonaws.com)
13 20115 [iot_threa] [ERROR][NET][20115] Failed to establish new connection. Socket status: -1.
14 20424 [iot_threa] [ERROR][MQTT][20424] Failed to establish new MQTT connection, error NETWORK ERROR.
15 20433 [iot_threa] [ERROR][DEMO][20433] MQTT CONNECT returned error NETWORK ERROR.
16 20441 [iot_threa] [INFO ][MQTT][20441] MQTT library cleanup done.
17 20447 [iot_threa] [ERROR][DEMO][20447] Error running demo.
Interface 0 IP address : 192.168.90.185
LwIP_DHCP: dhcp stop.
Deinitializing WIFI ...
18 20586 [iot_threa] [INFO ][INIT][20586] SDK cleanup done.
19 20591 [iot_threa] [INFO ][DEMO][20591] -------DEMO FINISHED-------
```


All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.

9.4 OTA Signature Check Fail

If the signature verification fail at the end of firmware download, it means the integrity check fail.

```
12594 497296 [OTA Agent T] [INFO] Current State=[WaitingForFileBlock], Event=[ReceivedFileBlock], New state=[WaitingForFileBlock]
12595 497307 [OTA Agent T] [INFO] Received valid file block: Block index=1050, Size=4096
12596 497315 [OTA Agent T] [INFO] otaPal_WriteBlock
12597 497320 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] C->fileSize 4304896, iOffset: 0x41a000: iBlockSize: 0x1000
12598 497331 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] ota_len:4304896, cur_block:1050
12599 497339 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] LAST image data arrived
12600 497395 [OTA Agent T] [INFO] [prvPAL_WriteBlock_amebaPro2] Write bytes: read_bytes 4096, ulBlockSize 4096
12601 497404 [OTA Agent T] [INFO] Received final block of the update.
12602 497410 [OTA Agent T] [INFO] otaPal_CloseFile
12603 497415 [OTA Agent T] [INFO] [prvPAL_CloseFile_amebaPro2] Authenticating and closing file.
12604 497423 [OTA Agent T] [INFO] [prvPAL_CheckFileSignature_amebaPro2] Started sig-sha256-ecdsa signature verification, file:
12605 497435 [OTA Agent T] [INFO] [prvPAL_ReadAndAssumeCertificate_amebaPro2] Assume Cert - No such file: . Using header file
12606 497446 [OTA Agent T] [INFO] [prvSignatureVerificationUpdate_amebaPro2] fw2 address will be upgraded
12607 497980 [iot_thread] [INFO]  Received: 1051   Queued: 1051   Processed: 1050   Dropped: 0
12608 498095 [OTA Agent T] [INFO] [prvSignatureVerificationUpdate_amebaPro2] Signature Verification Update done.
12609 498278 [OTA Agent T] [INFO] prvPAL_SetPlatformImageState_amebaPro2
12610 498332 [OTA Agent T] [ERROR] [prvPAL_CloseFile_amebaPro2] Failed to pass sig-sha256-ecdsa signature verification: 227.
12611 498341 [OTA Agent T] [INFO] prvPAL_SetPlatformImageState_amebaPro2
12612 498396 [OTA Agent T] [ERROR] Failed to close the OTA file: Error=(OtaPalSignatureCheckFailed:0xe3000000)
12613 498404 [OTA Agent T] [ERROR] Failed to ingest data block, rejecting image: ingestDataBlock returned error: OtaErr_t=-2
```

Please check the **otapalconfigCODE_SIGNING_CERTIFICATE** in “project/realtek_amebapro2_v0_example/src/aws_iot_freertos_lts/vendors/realtek/boards/amebaPro2/aws_demos/config_files/ota_demo_config.h”

Note: If there is any issue with OTA download, please feel free to open an issue on Github

All information provided in this document is subject to legal disclaimers.
© REALTEK 2022. All rights reserved.