WINC1500

WINC1500 AWS IoT Demo with RSA

Introduction

This application note provides a description on, how to use the SAMW25 Xplained PRO or SAMD21 Xplained PRO with WINC module to communicate with the AMAZON AWS IoT Cloud with RSA.

The DEMO provides an example of MQTT publish/subscribe with AWS IoT. The SAMW25 Xplained PRO or SAMD21 Xplained PRO with WINC module is configured as either Publisher (while the AWS Console MQTT client is the Subscriber) or Subscriber (while the AWS Console MQTT client is the Publisher).

Figure 1. SAMW25 XPRO

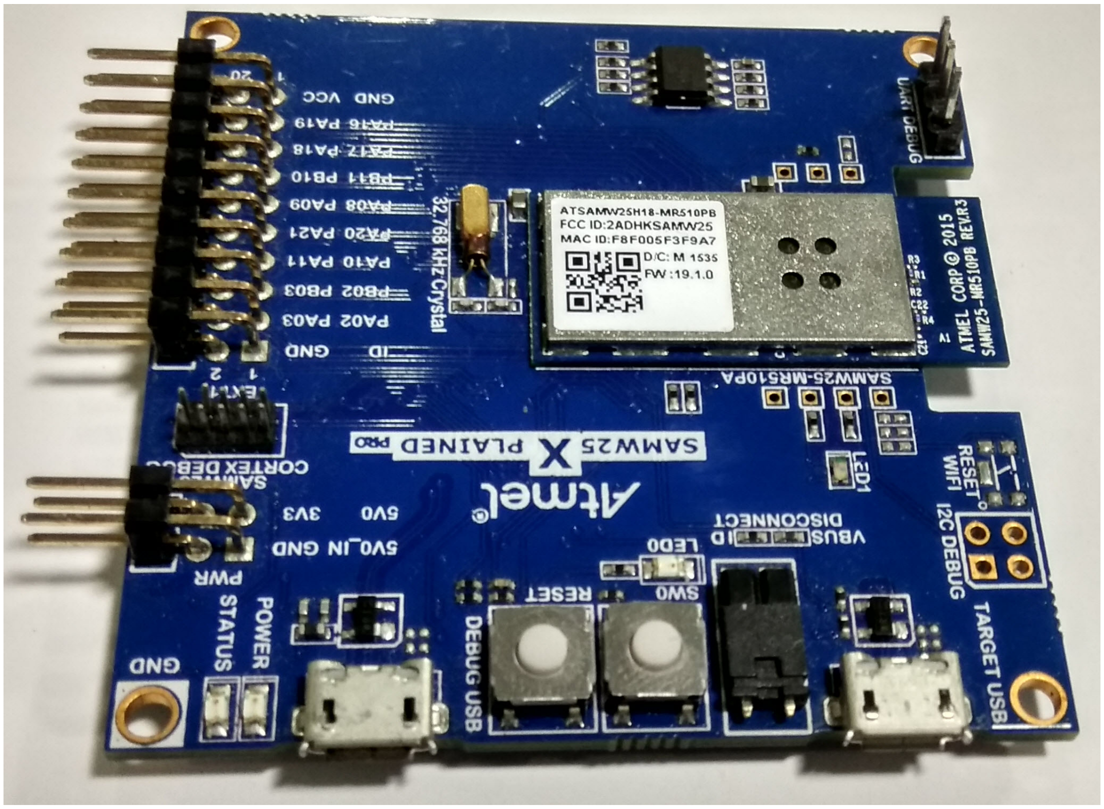


Figure 2. SAMD21 XPRO WITH WINC1500 CONNECTED ON EXT1

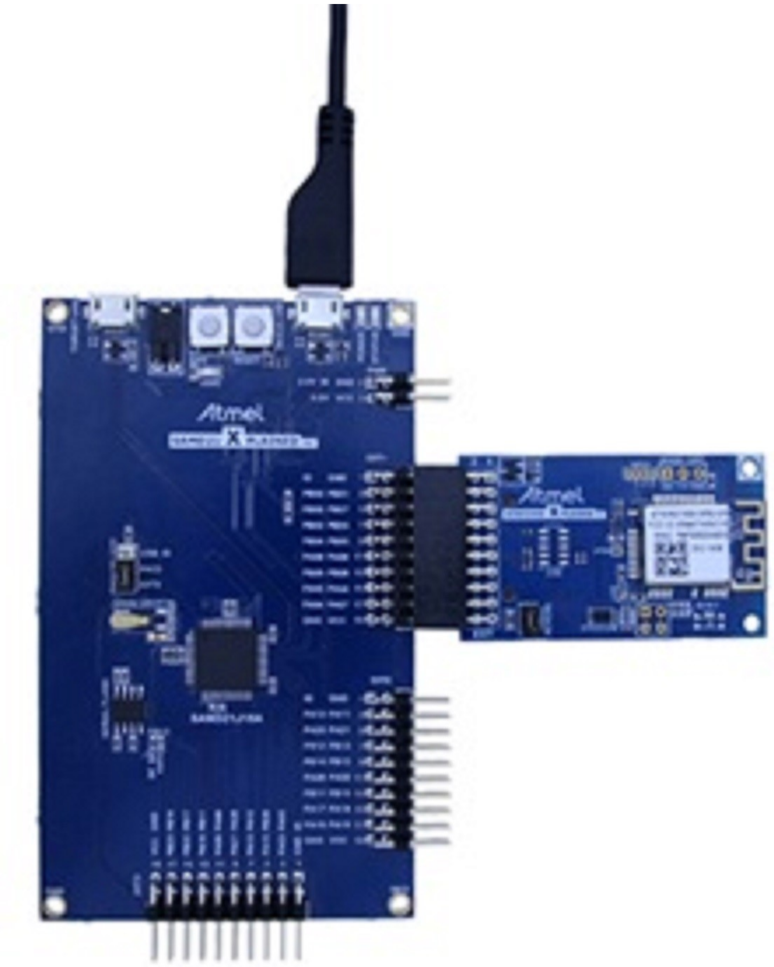


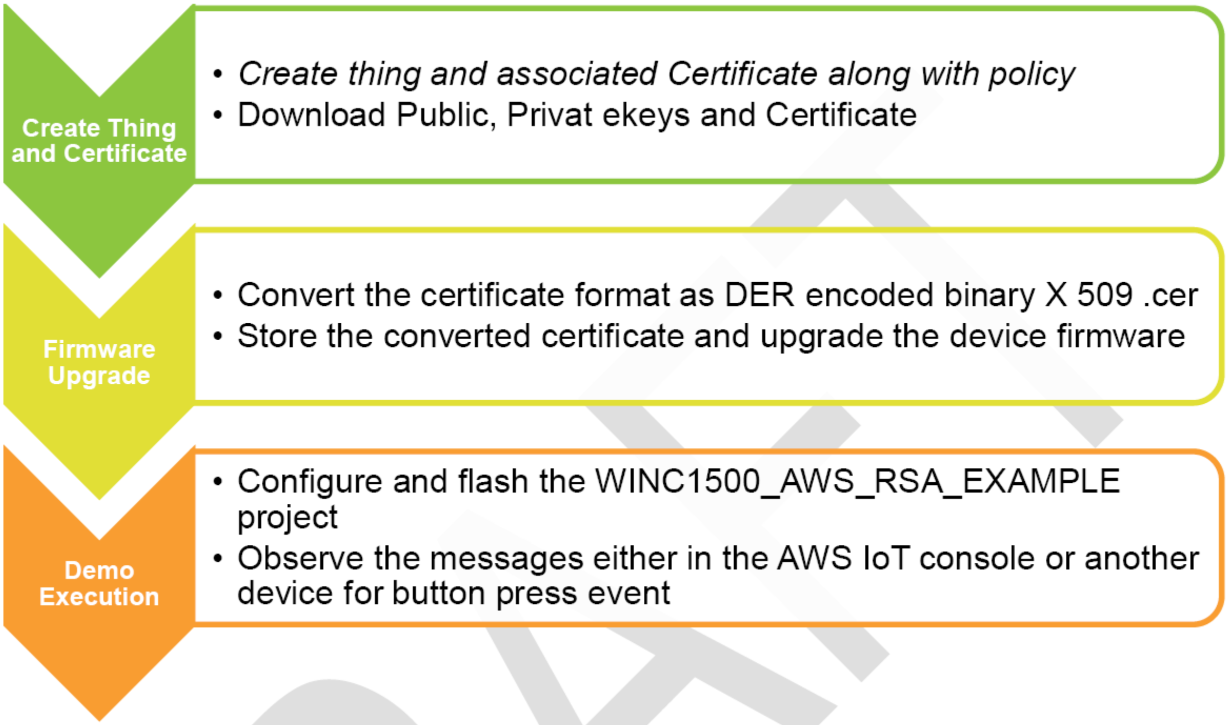
Table of Contents

1. Getting Started

1.1 Demo Application Flow

The sequence of activities to perform the AWS IoT demo application work with the RSA certificate are provided below.

Figure 1-1. AWS IoT DEMO APPLICATION FLOW



Create Thing and Certificate

- *Create thing and associated Certificate along with policy*
- Download Public, Privat ekeys and Certificate

Firmware Upgrade

- Convert the certificate format as DER encoded binary X 509 .cer
- Store the converted certificate and upgrade the device firmware

Demo Execution

- Configure and flash the WINC1500_AWS_RSA_EXAMPLE project
- Observe the messages either in the AWS IoT console or another device for button press event

Note: The details of each activity, and the preferred and required configuration values are provided in furthur sections.

1.2 Prerequisites

A valid Amazon AWS IoT account

Hardware Prerequisites:

- 2- SAMD21-XPRO Evaluation kit + WINC (WINC1500) module or
- 2- SAMW25 –XPRO Evaluation kit
- Micro-USB cable (Type A/Micro B)

Software Prerequisites:

- WINC1500 Release 19.5.2 Atmel Studio 7

2. AWS IoT Account Setup

This chapter demonstrates the setting up of the AWS IoT Account and various steps involving in registering and activating a device.

Figure 2-1. AWS IoT ACCOUNT SETUP

AWS IoT

Easily and securely connect devices to the cloud.
Reliably scale to billions of devices and trillions of messages.

2.1 Sign in to the AWS IoT console

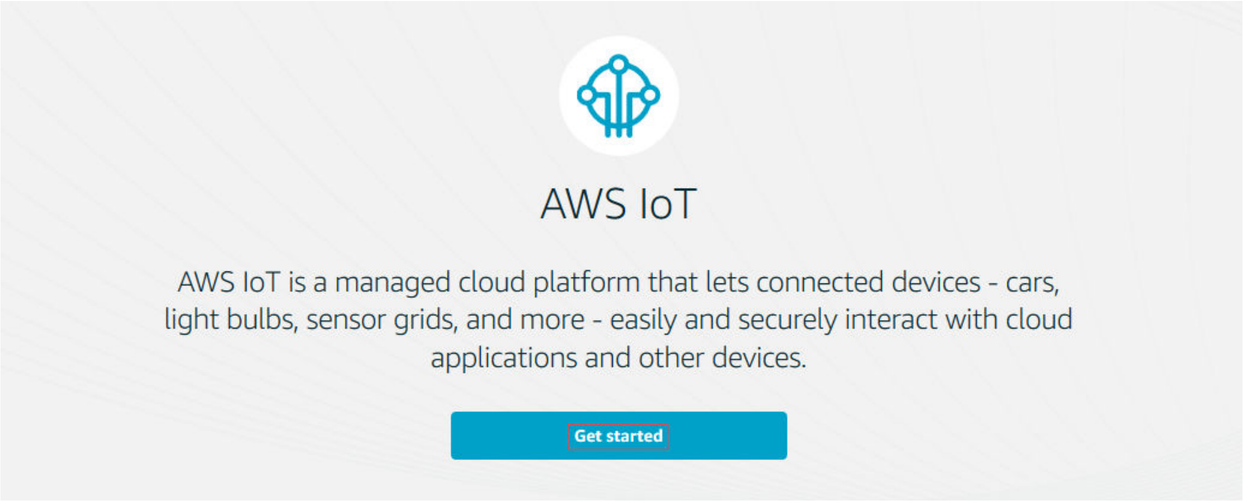
Sign in to the AWS IOT console.

Note: If you do not have an AWS account, create the account.

To create an Amazon Web Services (AWS) account:

1. Open the AWS home page and select **Create an AWS Account**.
2. Follow the online instructions. A part of the sign-up procedure involves receiving a phone call and entering a PIN using user's phone keypad.
3. Sign in to the AWS Management console and open the AWS IoT console.
4. On the **Welcome** page, select **Get started**.
   Note: First time user of AWS IoT console finds the **Welcome to the AWS IoT Console** page.

Figure 2-2. GETTING STARTED WITH AWS IoT CONSOLE



AWS IoT

AWS IoT is a managed cloud platform that lets connected devices - cars, light bulbs, sensor grids, and more - easily and securely interact with cloud applications and other devices.

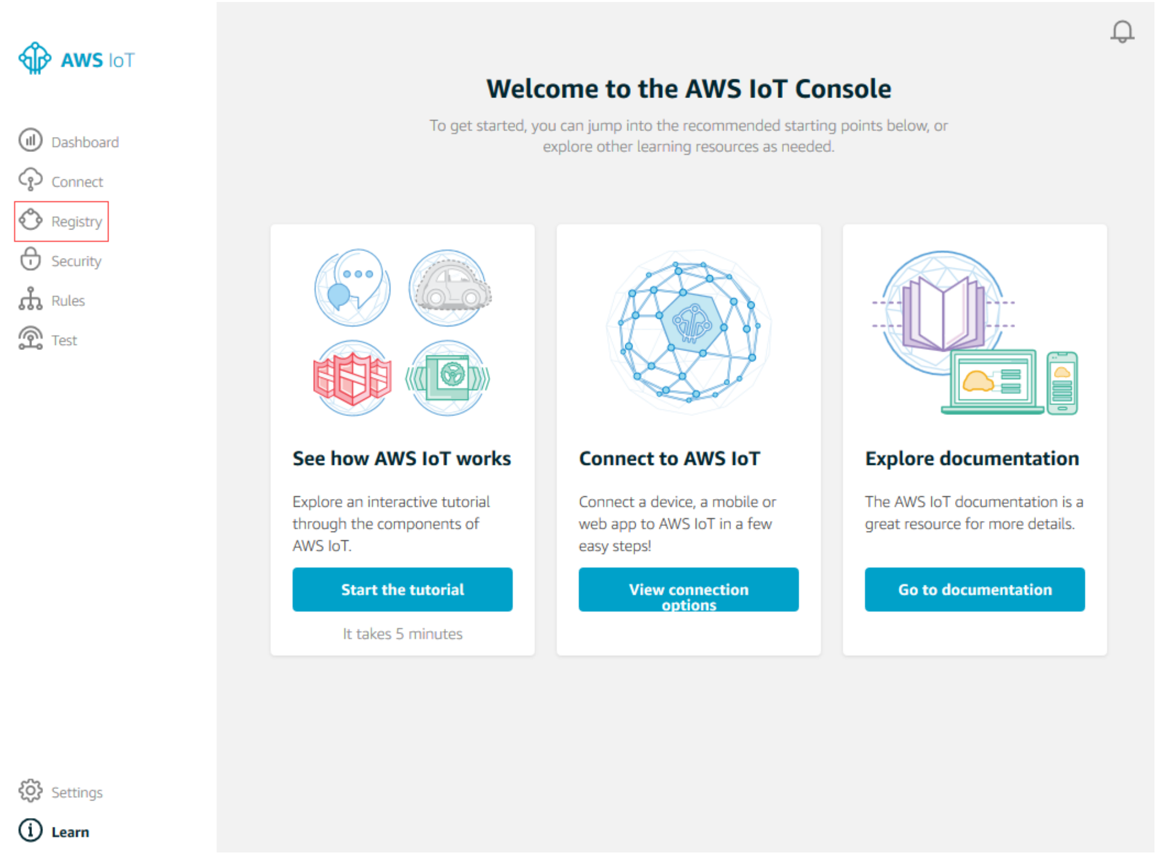
2.2 Register a Device in the Thing Registry

In the Thing registry, the devices connected to AWS IoT are represented by things. The Thing registry allows to keep a record of all devices that are connected to an AWS IoT account.

To register a device in the thing registry:

1. On the **Welcome to the AWS IoT Console** page in the left navigation area, choose **Registry** to expand the choices, and then select **Things**.

Figure 2-3. WELCOME PAGE



AWS IoT

Dashboard
Connect
Registry
Security
Rules
Test

Settings
Learn

Welcome to the AWS IoT Console

To get started, you can jump into the recommended starting points below, or explore other learning resources as needed.

See how AWS IoT works

Explore an interactive tutorial through the components of AWS IoT.

Start the tutorial

It takes 5 minutes

Connect to AWS IoT

Connect a device, a mobile or web app to AWS IoT in a few easy steps!

View connection options

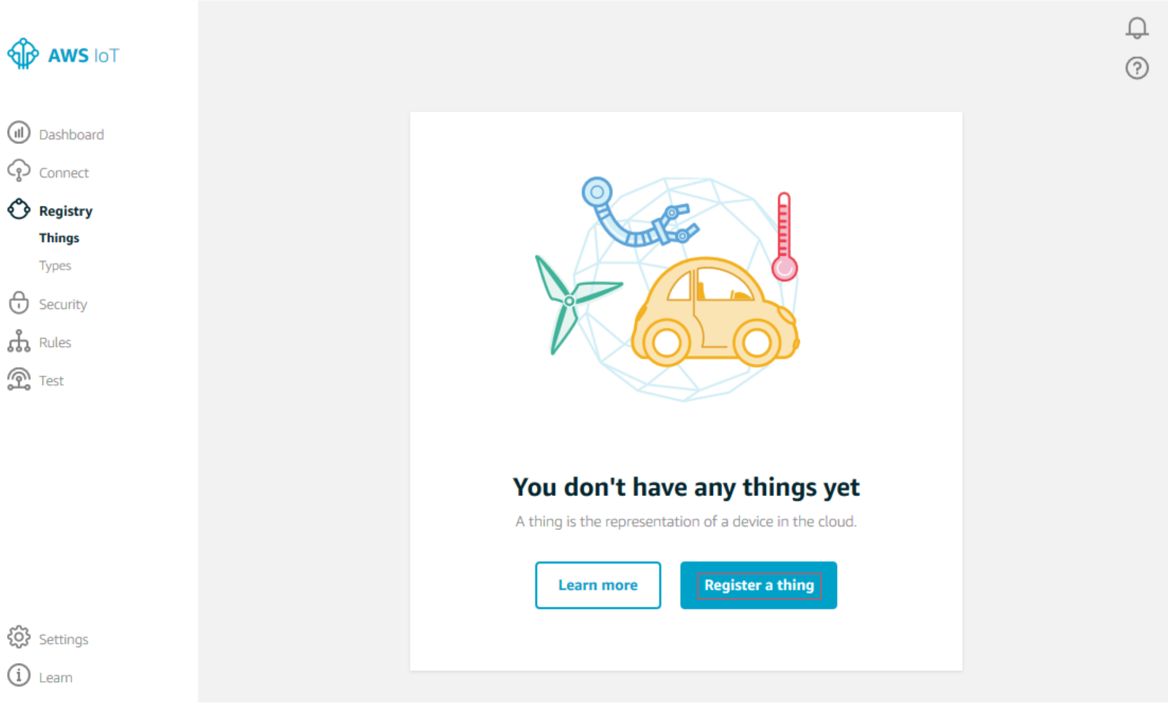
Explore documentation

The AWS IoT documentation is a great resource for more details.

Go to documentation

2. On the **You don't have any things yet** page, select **Register a thing**.

Figure 2-4. REGISTERING A THING



AWS IoT

Dashboard
Connect
Registry
Things
Types
Security
Rules
Test

Settings
Learn

You don't have any things yet

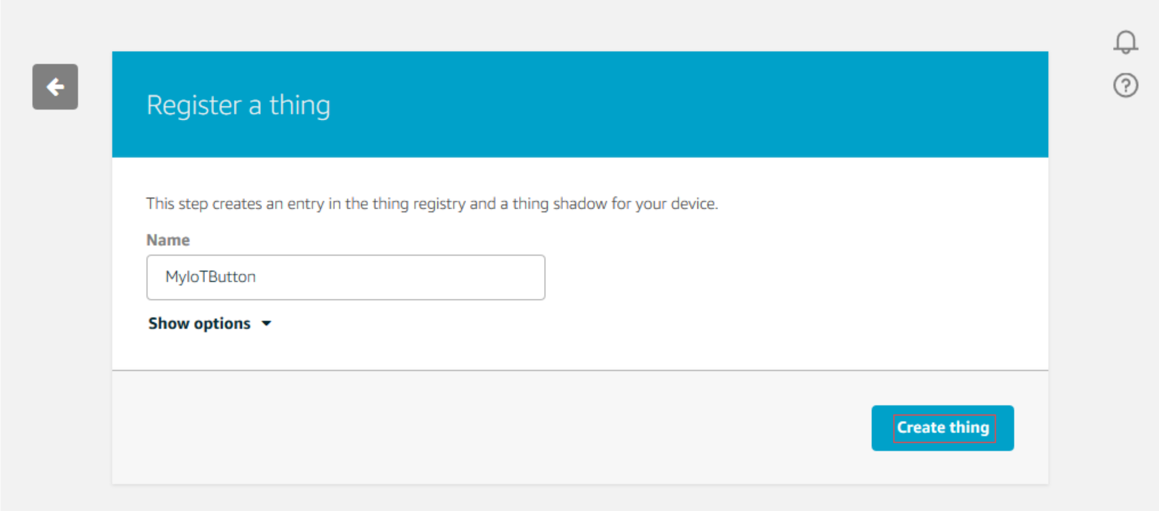
A thing is the representation of a device in the cloud.

Learn more

Register a thing

3. On the **Register a thing** page in the **Name** field, type a name for your device. Select **Create thing** to add device to the Thing registry.

Figure 2-5. CREATE A THING



Register a thing

This step creates an entry in the thing registry and a thing shadow for your device.

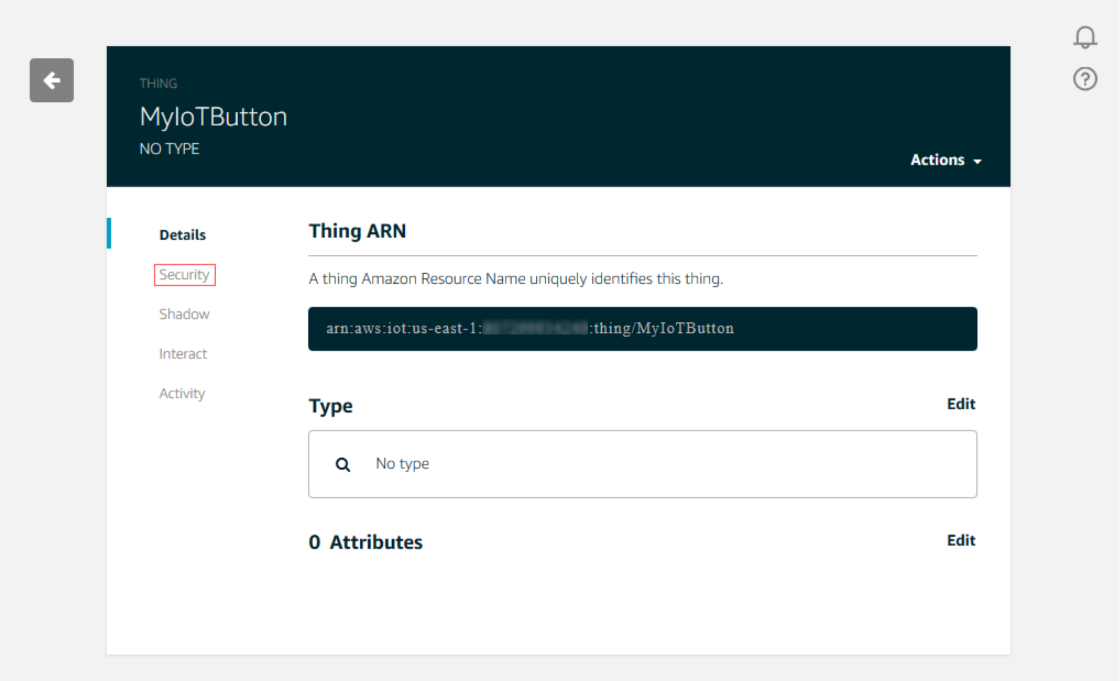
Name: MyIoTButton

2.3 Create and Activate a Device Certificate

The communication between the device and AWS IoT are protected through the use of X.509 certificates. The AWS IoT generates a certificate or the user can use their own X.509 certificate. This demonstration assumes that AWS IoT generates the X.509 certificate. The certificates must be activated prior to use.

1. On the **Details** page in the left navigation area, choose **Security**.

Figure 2-6. DETAILS PAGE



THING

MyIoTButton

NO TYPE

Thing ARN

A thing Amazon Resource Name uniquely identifies this thing.

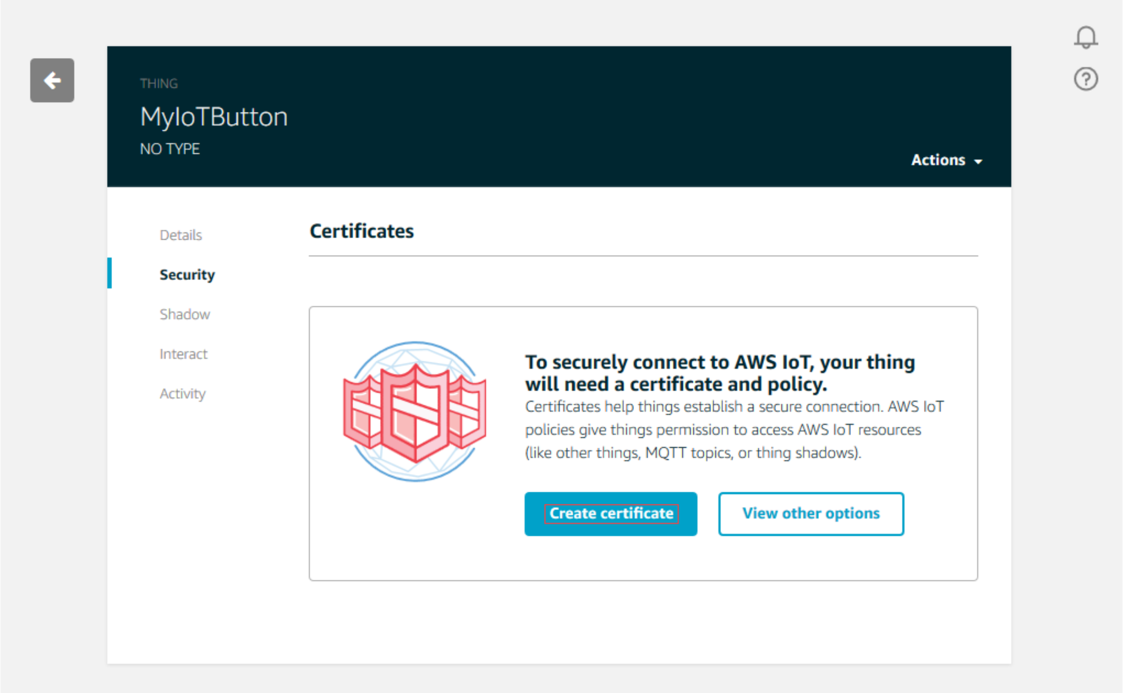
arn:aws:iot:us-east-1:[redacted]:thing/MyIoTButton

Type: No type

0 Attributes

2. On the **Certificates** page, select **Create certificate**.

Figure 2-7. CERTIFICATES PAGE



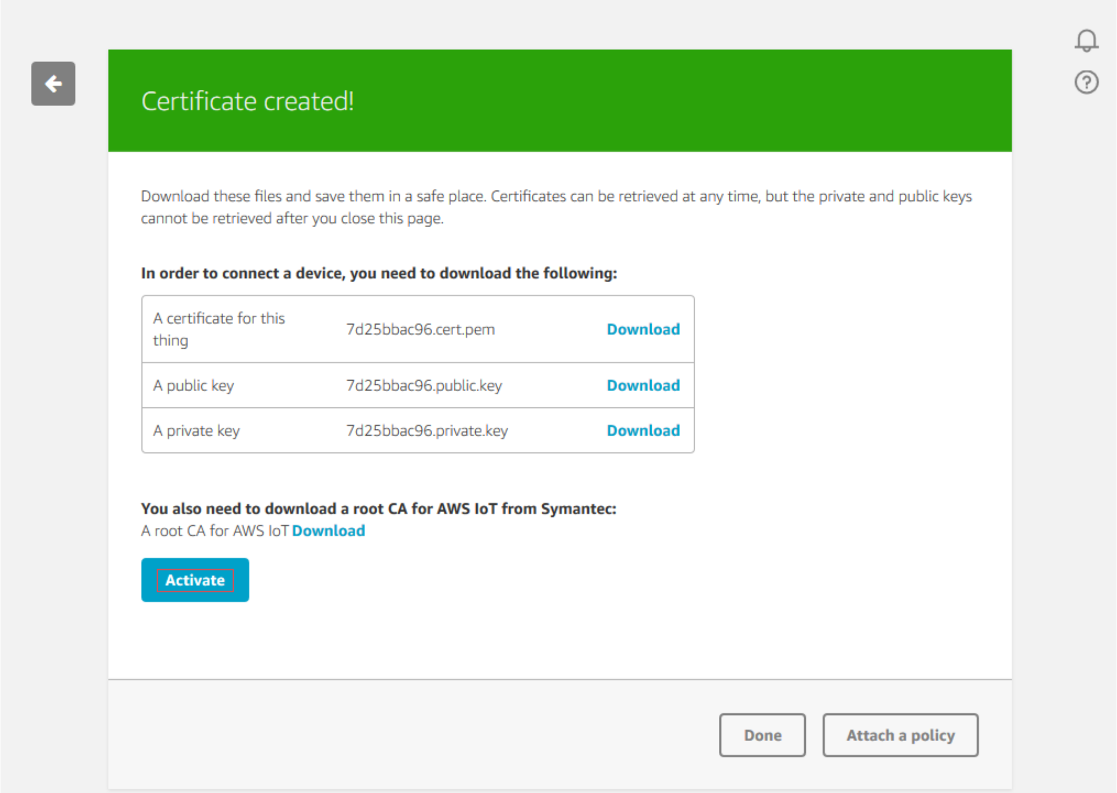
To securely connect to AWS IoT, your thing will need a certificate and policy.

Certificates help things establish a secure connection. AWS IoT policies give things permission to access AWS IoT resources (like other things, MQTT topics, or thing shadows).

3. On the **Certificate created** page, select **Download** for the certificate, public key, private key, and the root CA for AWS IoT.
   Ensure to save each of these downloads to PC, and then select **Activate** to continue.
   Note: The downloaded certificate must be converted from .crt format to .cer format.

   Rename the "certificate and key" as "atmelwinc.cer and atmelwinc.key" and place them in the **src/tls_cert_store** folder path of firmware upgrade project for later use.

Figure 2-8. CERTIFICATE CREATED PAGE



Certificate created!

Download these files and save them in a safe place. Certificates can be retrieved at any time, but the private and public keys cannot be retrieved after you close this page.

In order to connect a device, you need to download the following:

A certificate for this thing	7d25bbac96.cert.pem	Download
A public key	7d25bbac96.public.key	Download
A private key	7d25bbac96.private.key	Download

You also need to download a root CA for AWS IoT from Symantec:
A root CA for AWS IoT **Download**

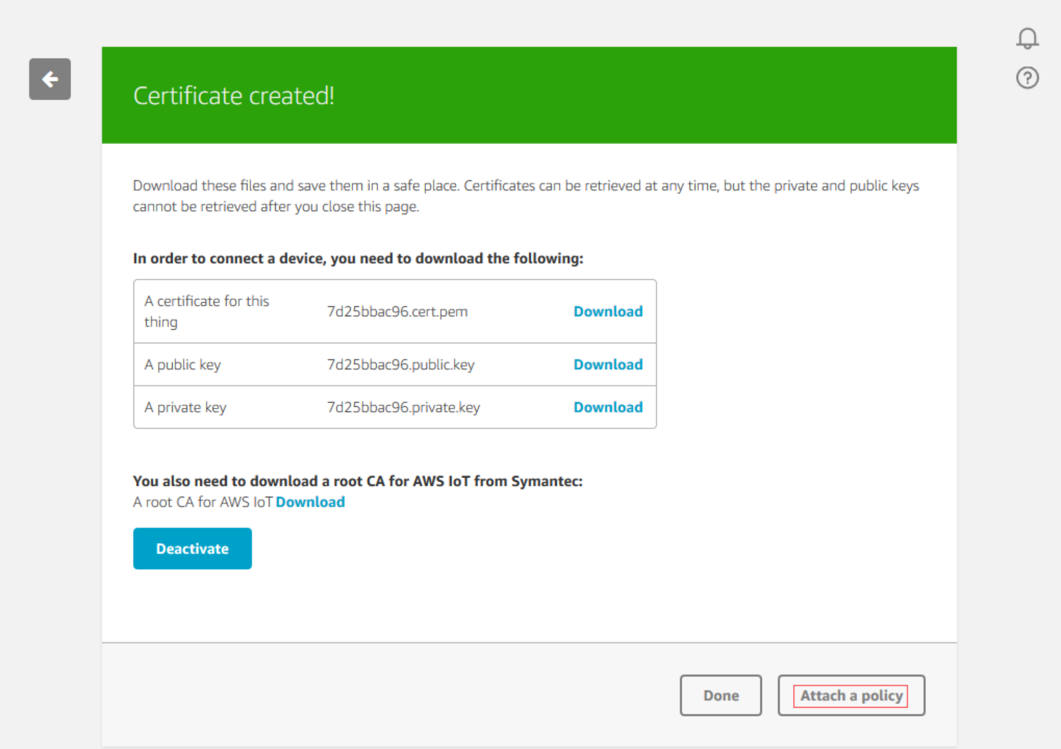
2.4 Attach an AWS IoT Policy

The X.509 certificates are used to authenticate the device with the AWS IoT. The AWS IoT policies are used to authorize the device to perform AWS IoT operations, such as Subscribing or Publishing to MQTT topics. The device will present its certificate, while sending messages to AWS IoT. To allow the device to perform AWS IoT operations, user must create an AWS IoT policy and attach it to the device certificate.

To create an AWS IoT policy:

1. On the **Certificate created** page, select **Attach a policy**.

Figure 2-9. CERTIFICATE CREATED PAGE



Certificate created!

Download these files and save them in a safe place. Certificates can be retrieved at any time, but the private and public keys cannot be retrieved after you close this page.

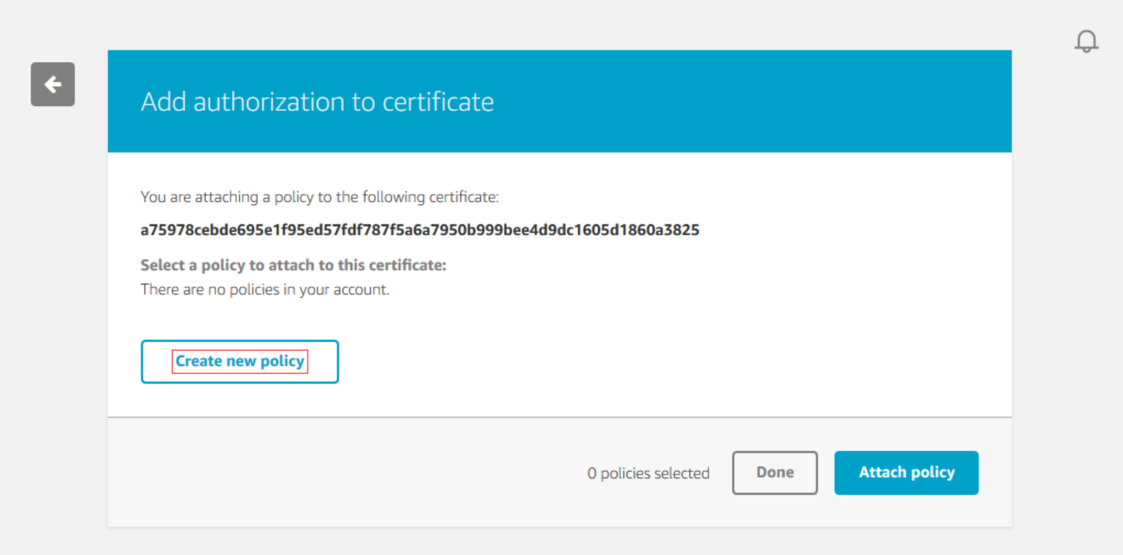
In order to connect a device, you need to download the following:

A certificate for this thing	7d25bbac96.cert.pem	Download
A public key	7d25bbac96.public.key	Download
A private key	7d25bbac96.private.key	Download

You also need to download a root CA for AWS IoT from Symantec:
A root CA for AWS IoT **Download**

2. On the **Add authorization to certificate** page, select **Create new policy**.

Figure 2-10. ADDING AUTHORIZATION TO CERTIFICATE PAGE



Add authorization to certificate

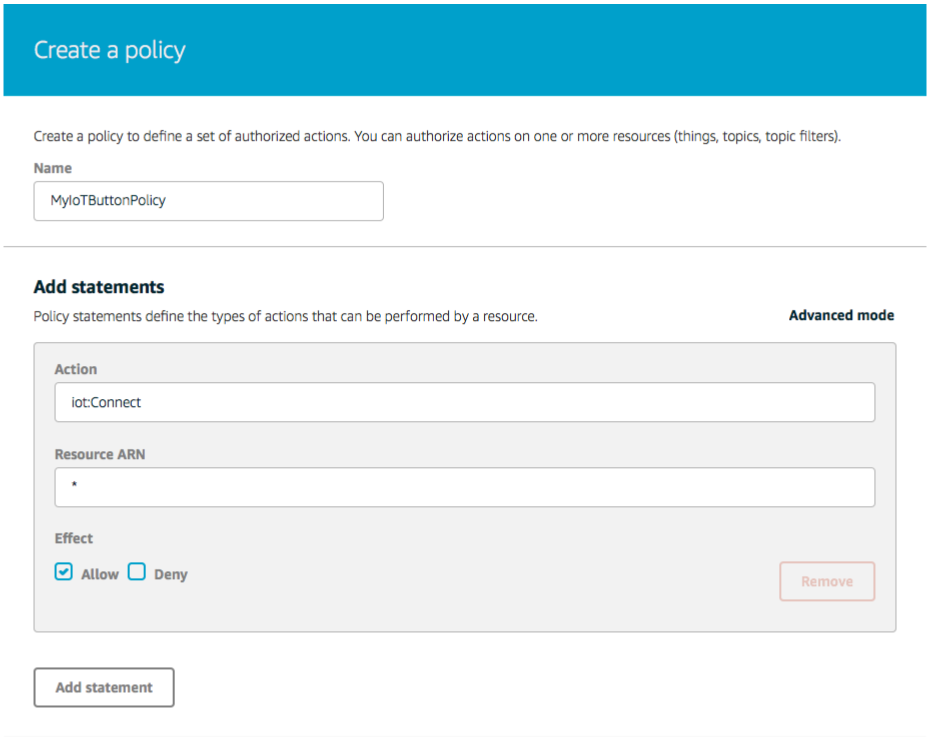
You are attaching a policy to the following certificate:

a75978cebde695e1f95ed57fdf787f5a6a7950b999bee4d9dc1605d1860a3825

Select a policy to attach to this certificate:
There are no policies in your account.

3. On the **Add authorization to certificate** page, set the following:
   - In the **Name** field, type a name for the policy (for example, "myPolicy")
   - In the **Action** field, type **iot:Connect**
   - In the **Resource ARN** field, type *
   - Select **Allow** check box. This allows all clients to connect with AWS IoT and also allows the device to publish messages to the specified topic.

Figure 2-11. CONNECTING TO AWS IoT



Create a policy

Create a policy to define a set of authorized actions. You can authorize actions on one or more resources (things, topics, topic filters).

Name: MyIoTButtonPolicy

Add statements

Policy statements define the types of actions that can be performed by a resource.

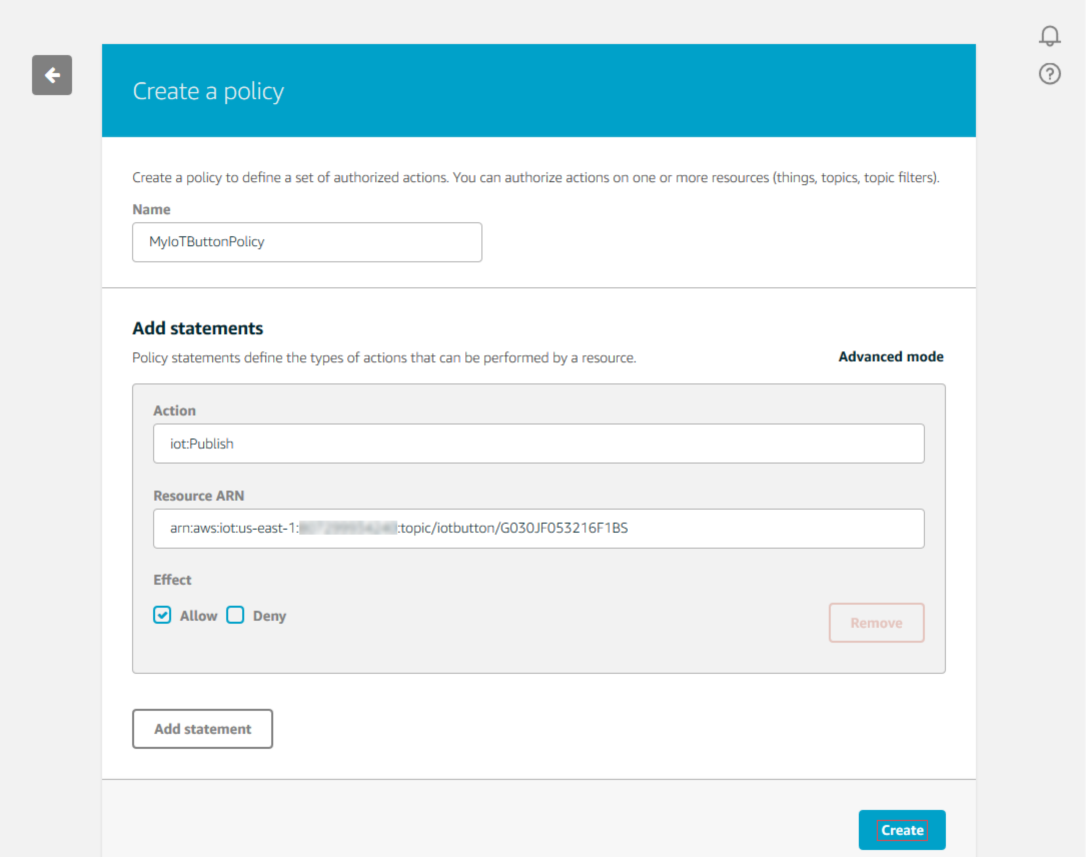
Action: iot:Connect

Resource ARN: *

Effect: ☑ Allow □ Deny

4. After entering all the information for the policy, click "Create".

Figure 2-12. CREATING A POLICY



Create a policy

Create a policy to define a set of authorized actions. You can authorize actions on one or more resources (things, topics, topic filters).

Name: MyIoTButtonPolicy

Add statements

Policy statements define the types of actions that can be performed by a resource.

Action: iot:Publish

Resource ARN: arn:aws:iot:us-east-1:[redacted]:topic/iotbutton/G030JF053216F1BS

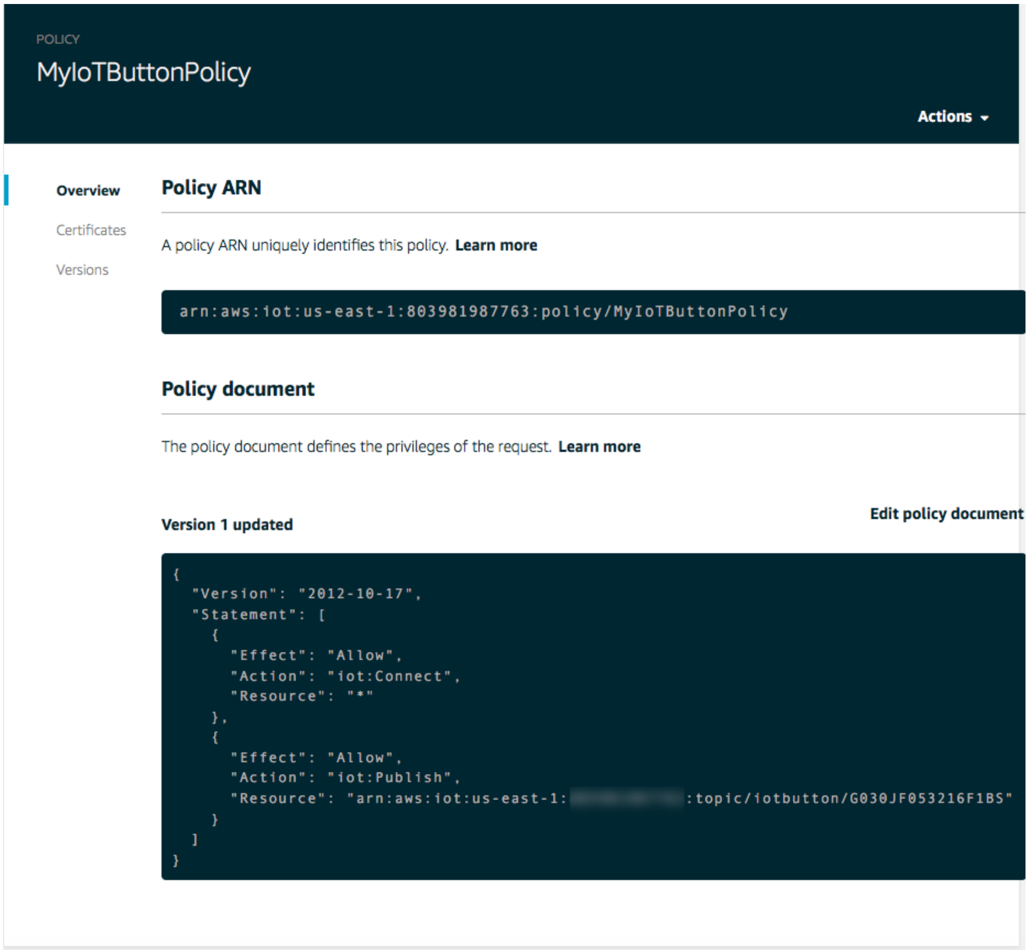
Effect: ☑ Allow □ Deny

2.5 Attach an AWS IoT Policy to a Device Certificate

After creating a AWS IoT policy, ensure to attach it with user device certificate. Attaching an AWS IoT policy with a certificate gives the device about the permissions specified in the policy.

1. On the **Overview** page for the policy in the left navigation area, select the left arrow to go to the AWS IoT **Policies** page.

Figure 2-13. POLICY ARN OVERVIEW PAGE



Policy ARN

A policy ARN uniquely identifies this policy. **Learn more**

```
arn:aws:iot:us-east-1:803981987763:policy/MyIoTButtonPolicy
```

Policy document

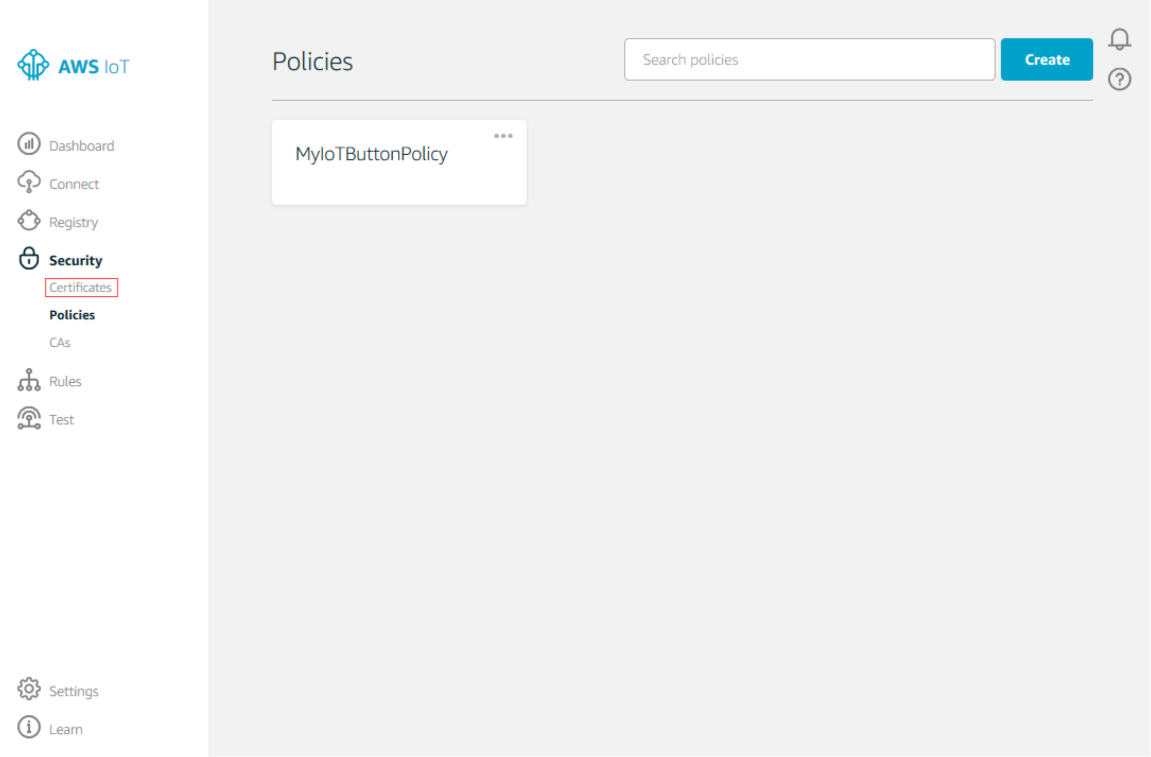
The policy document defines the privileges of the request. **Learn more**

Version 1 updated

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:Connect",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Publish",
      "Resource": "arn:aws:iot:us-east-1:[redacted]:topic/iotbutton/G030JF053216F1BS"
    }
  ]
}
```

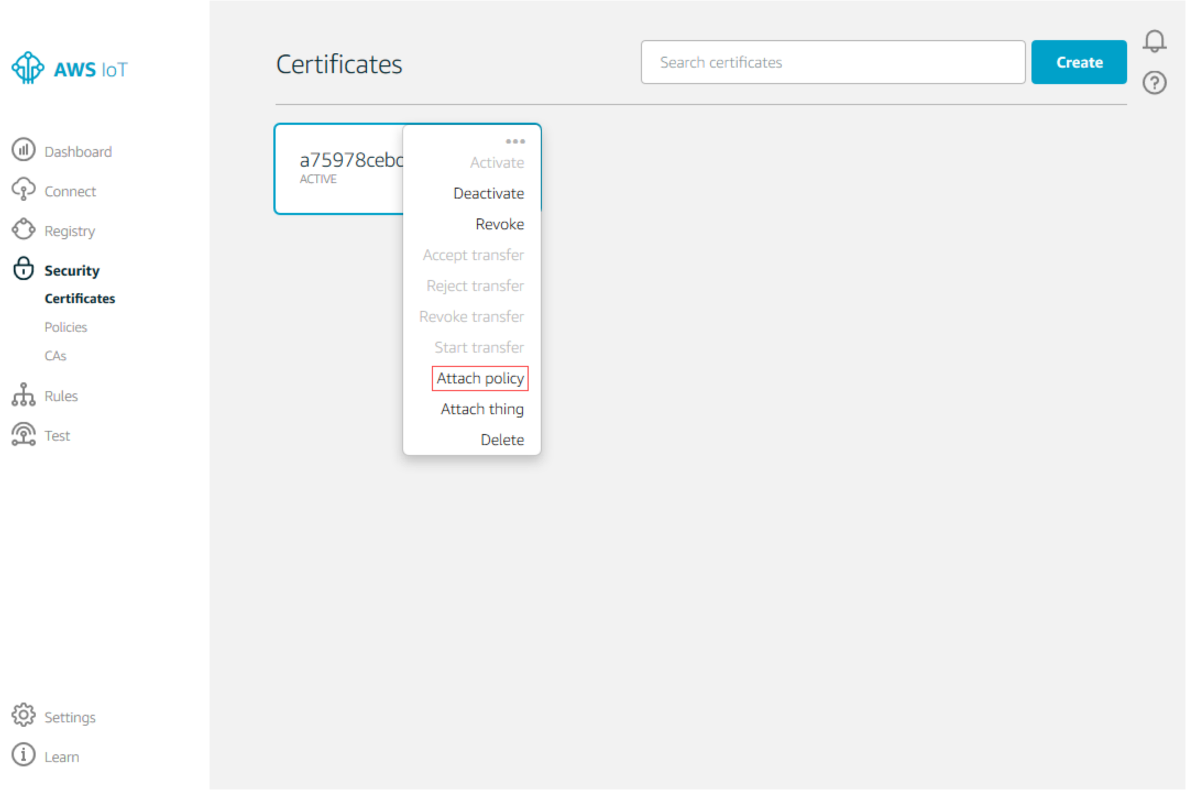
2. On the **Policies** page in the left navigation area, under **Security**, select **Certificates**.

Figure 2-14. POLICIES PAGE



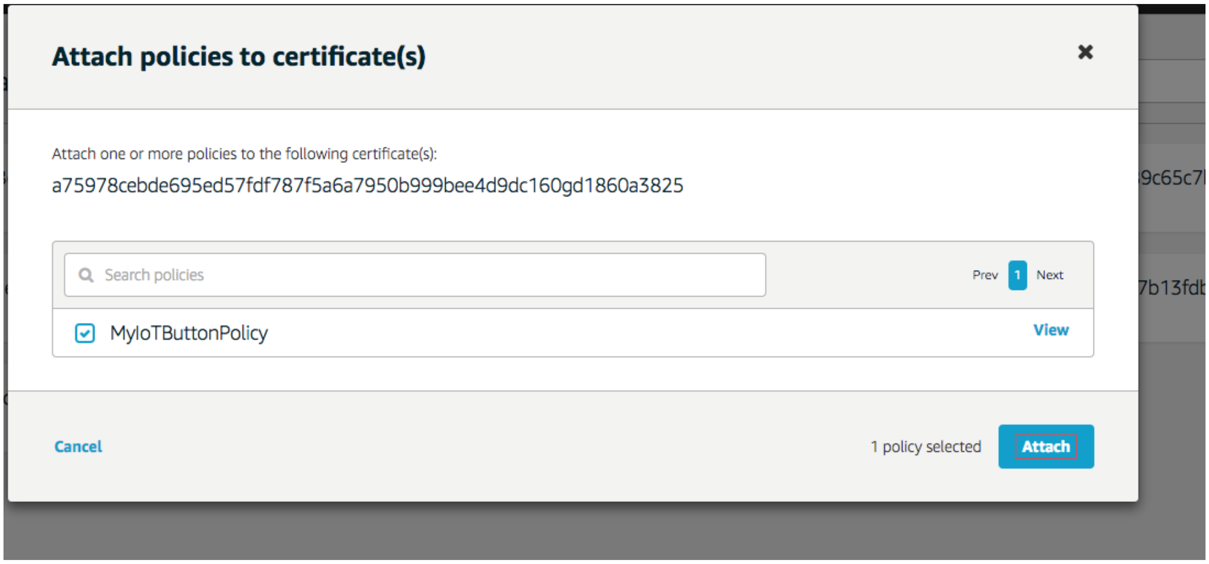
3. In the box for the certificate user created, choose **...** to open a drop-down menu, and then choose **Attach policy**.

Figure 2-15. CERTIFICATE PAGE



4. In the **Attach policies to certificate(s)** dialog box, select the check box next to the Policy created in the previous step, and then click **Attach**.

Figure 2-16. ATTACH POLICIES TO CERTIFICATE



Attach policies to certificate(s)

Attach one or more policies to the following certificate(s):

a75978cebde695ed57fdf787f5a6a7950b999bee4d9dc160gd1860a3825

☑ MyIoTButtonPolicy

1 policy selected

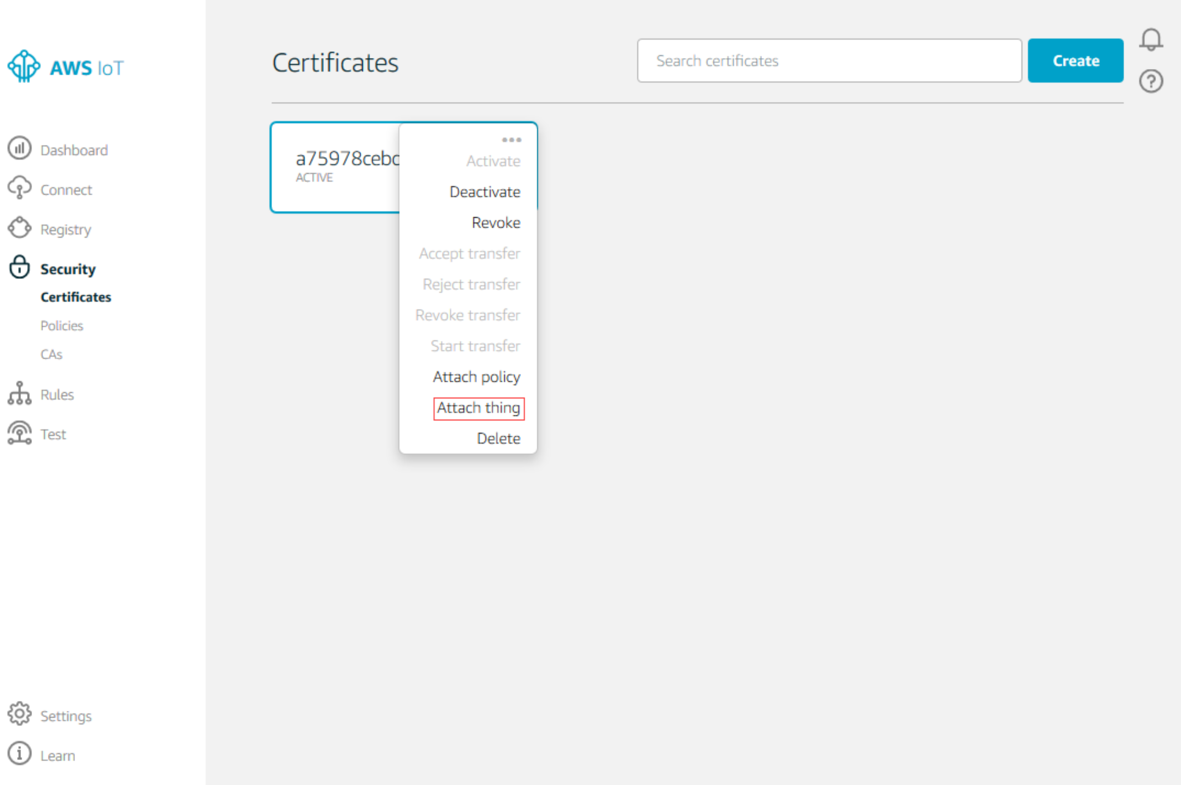
2.6 Attach a Certificate to a Thing

A device must have a certificate, private key and root CA certificate to authenticate with AWS IoT. It is also recommend to attach the device certificate to the thing that represents user device in AWS IoT. This allows to create AWS IoT policies that grant permissions based on certificates attached to user things. For more information. see Thing Policy Variables.

To attach a certificate to the thing representing device in the Thing registry:

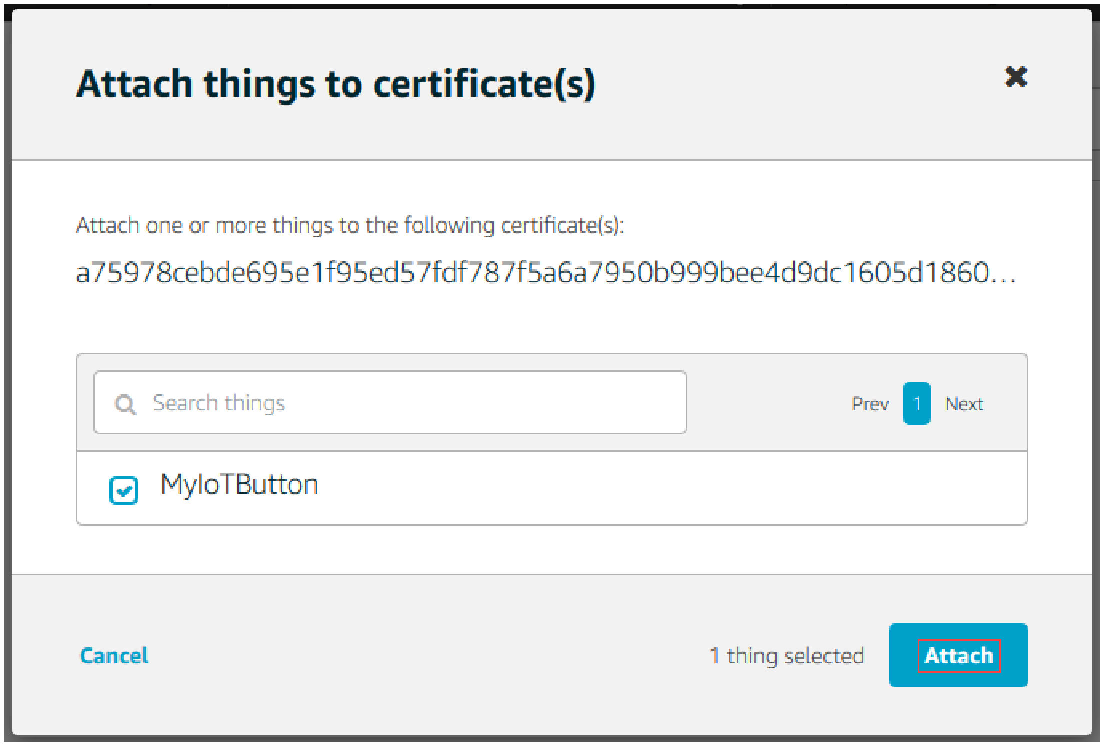
1. In the field of the certificate created, choose **...** to open a dropdown menu, and then choose **Attach thing**.

Figure 2-17. ATTACH A THING



2. In the **Attach things to certificate(s)** dialog box, select the check box next to Thing registered, and then click **Attach**.

Figure 2-18. ATTACH THINGS TO CERTIFICATE



Attach things to certificate(s)

Attach one or more things to the following certificate(s):

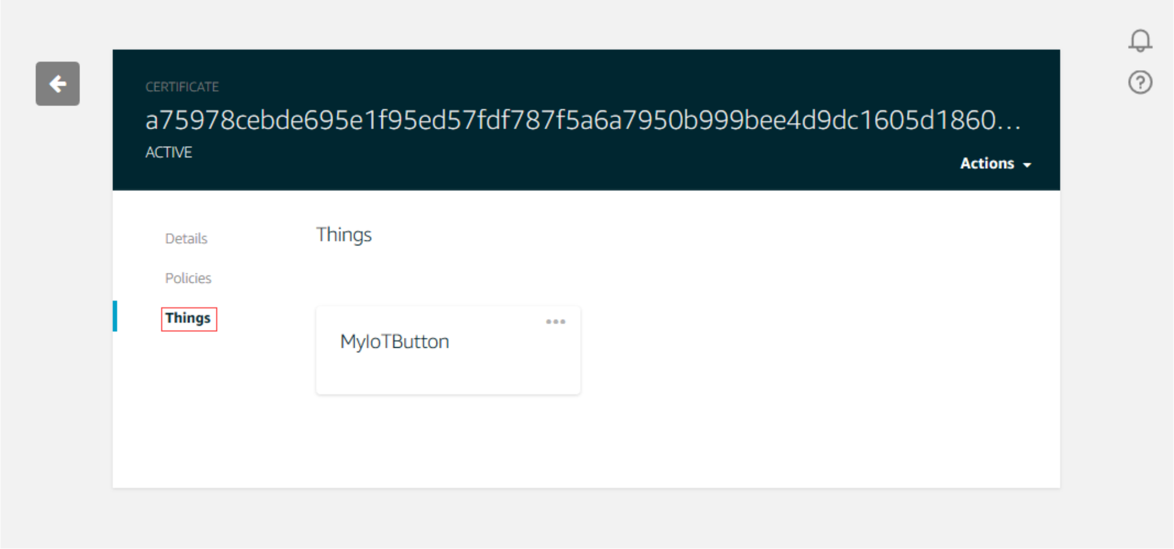
a75978cebde695e1f95ed57fdf787f5a6a7950b999bee4d9dc1605d1860…

☑ MyIoTButton

1 thing selected

3. To verify the thing is attached, select the box representing the certificate. On the **Details** page for the certificate in the left navigation area, choose **Things**.

Figure 2-19. THINGS PAGE



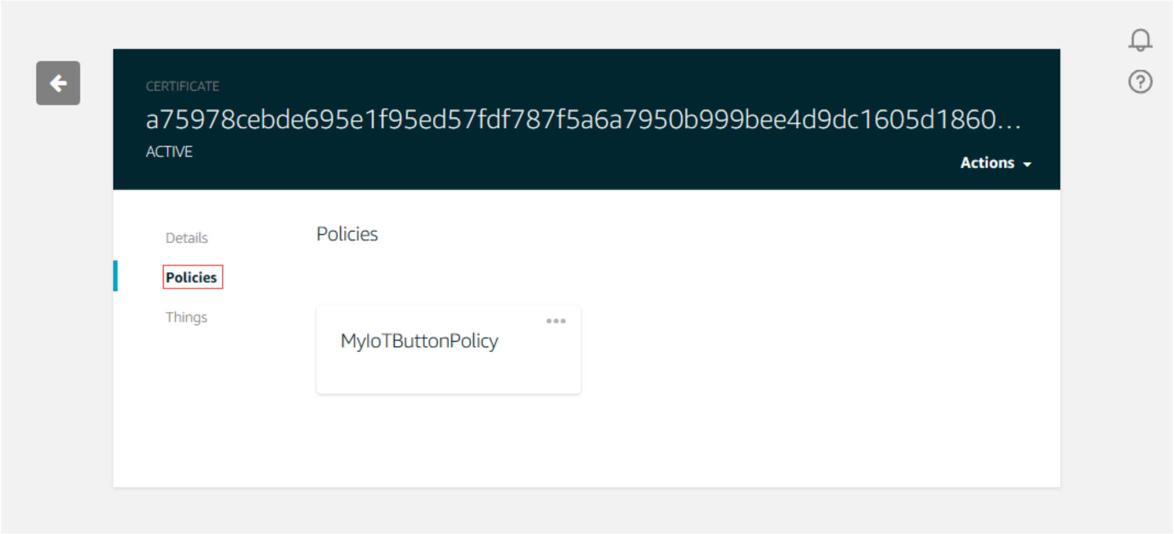
CERTIFICATE

a75978cebde695e1f95ed57fdf787f5a6a7950b999bee4d9dc1605d1860...

ACTIVE

4. To verify the policy is attached on the **Details** page for the certificate in the left navigation area, select **Policies**.

Figure 2-20. VERIFYING THE ATTACHED POLICY



CERTIFICATE

a75978cebde695e1f95ed57fdf787f5a6a7950b999bee4d9dc1605d1860...

ACTIVE

2.7 View Device MQTT Messages with the AWS IoT MQTT Client

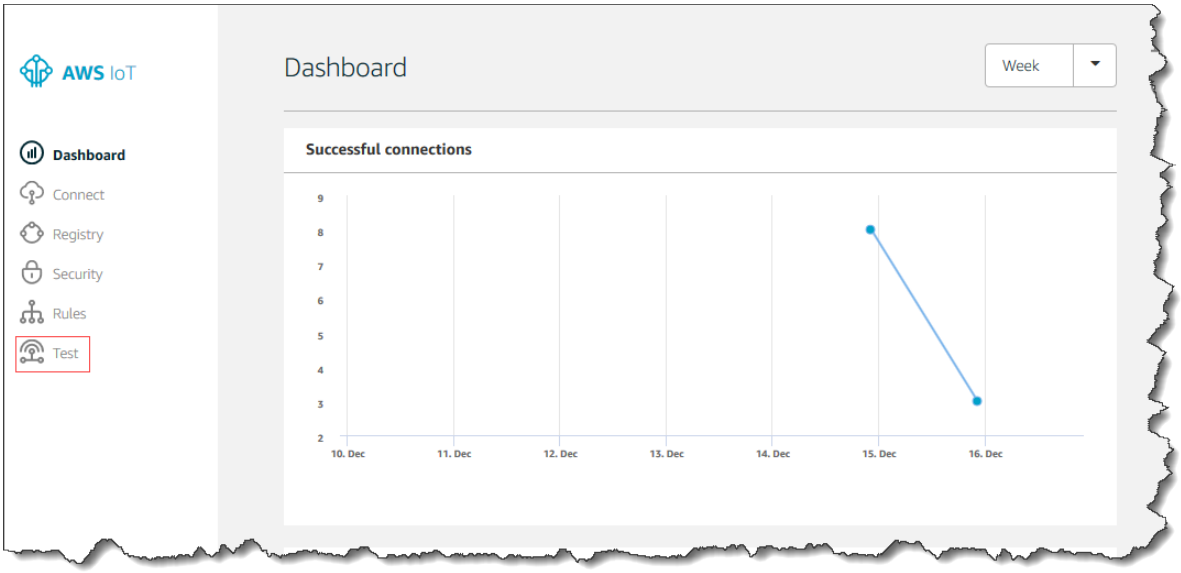
The user can use the AWS IoT MQTT client for better understanding the MQTT messages, sent by a device.

The devices publish MQTT messages on topics. The user can use the AWS IoT MQTT client to subscribe the topics for viewing these MQTT messages.

To view MQTT messages:

1. In the AWS IoT console in the left navigation area, select **Test**.

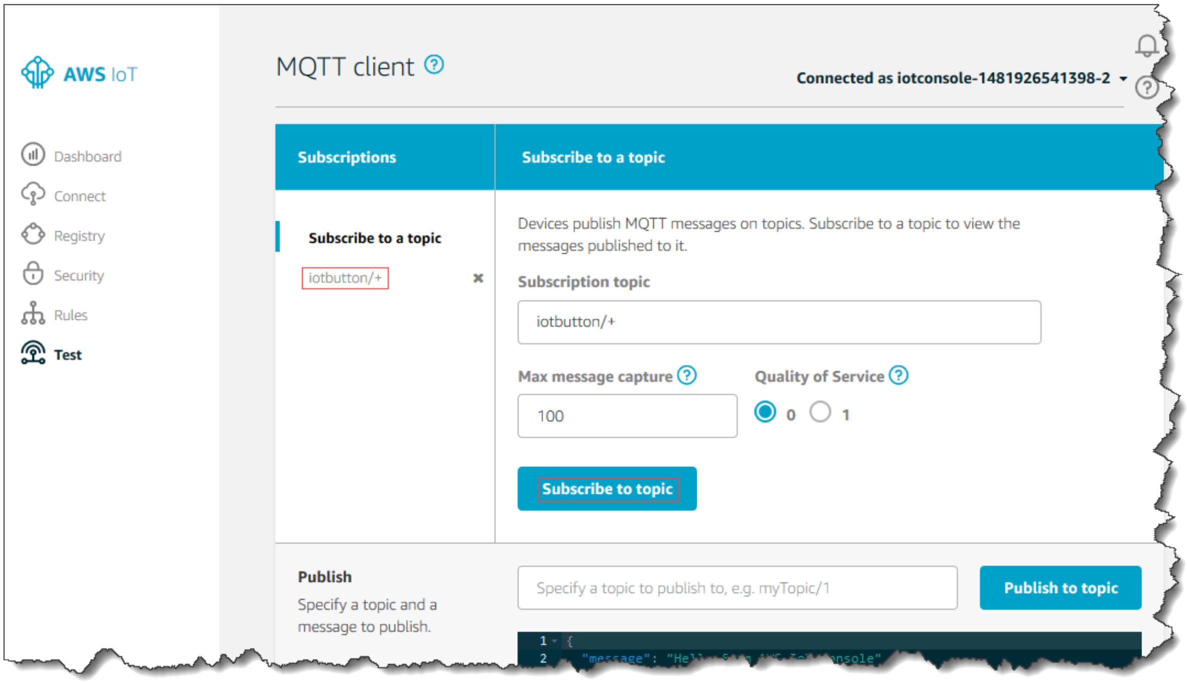
Figure 2-21. AWS IoT CONSOLE DASHBOARD



2. "Subscribe to a topic" on which the Thing publishes. In case of the AWS IoT button, the user can subscribe to **iotbutton/+**. In "Subscribe to a topic" in the "Subscription topic" field, type **iotbutton/+**, and then select **Subscribe to topic**.

   Note: The "Subscription topic" must appears under Subscriptions.

Figure 2-22. SUBSCRIPTIONS PAGE



Devices publish MQTT messages on topics. Subscribe to a topic to view the messages published to it.

Subscription topic: iotbutton/+

Max message capture: 100

Quality of Service: ● 0 ○ 1

Specify a topic and a message to publish.

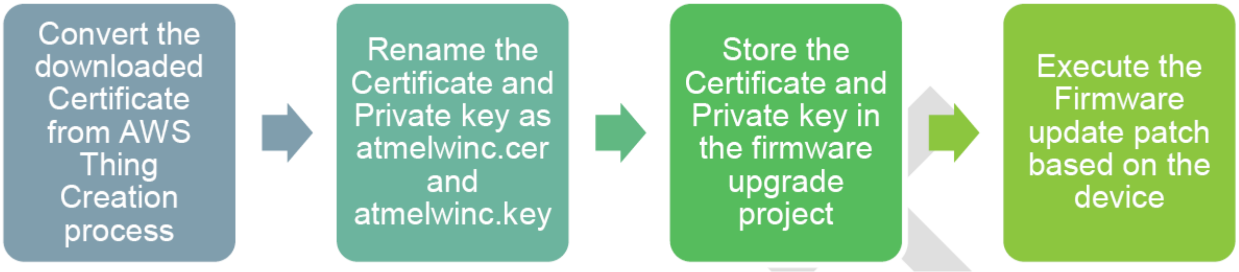
Specify a topic to publish to, e.g. myTopic/1

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

3. Programming Certificates

The Programming certificate from AWS IoT is downloaded in to the device for performing the AWS IoT connection with RSA.

Figure 3-1. PROGRAMMING CERTIFICATES PROCESS



Convert the downloaded Certificate from AWS Thing Creation process

Rename the Certificate and Private key as atmelwinc.cer and atmelwinc.key

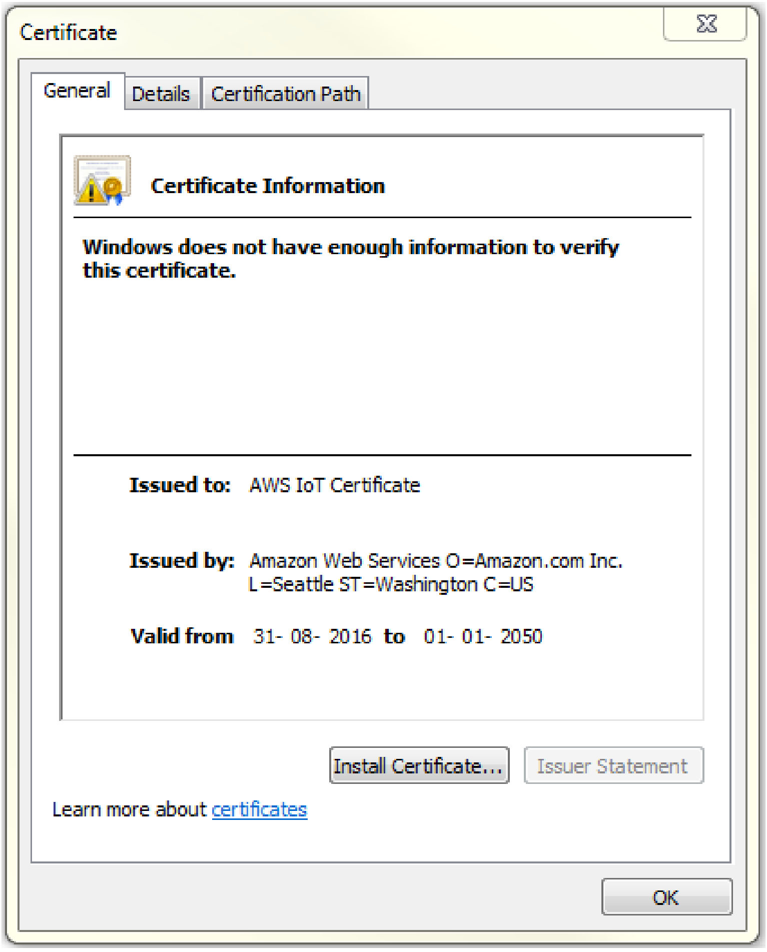
Store the Certificate and Private key in the firmware upgrade project

Execute the Firmware update patch based on the device

3.1 Certificate Conversion

1. Open the certificate file downloaded from AWS IoT (Certificate file generation, refer section Create and Activate a Device Certificate).

   Figure 3-2. DEVICE CERTIFICATE WINDOW



Certificate Information

Windows does not have enough information to verify this certificate.

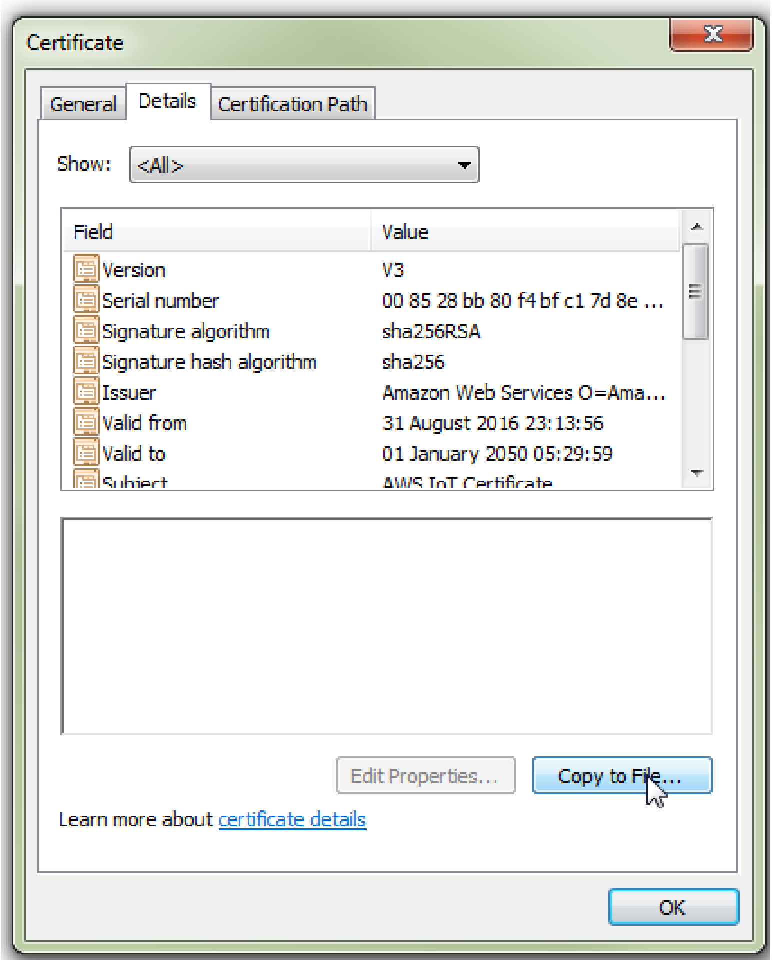
Issued to: AWS IoT Certificate

Issued by: Amazon Web Services O=Amazon.com Inc. L=Seattle ST=Washington C=US

Valid from 31- 08- 2016 **to** 01- 01- 2050

2. Open the "Details" tab and click "Copy to File", which displays "Certificate Export Wizard" dialogue box.

Figure 3-3. DETAILS PAGE IN CERTIFICATE WINDOW



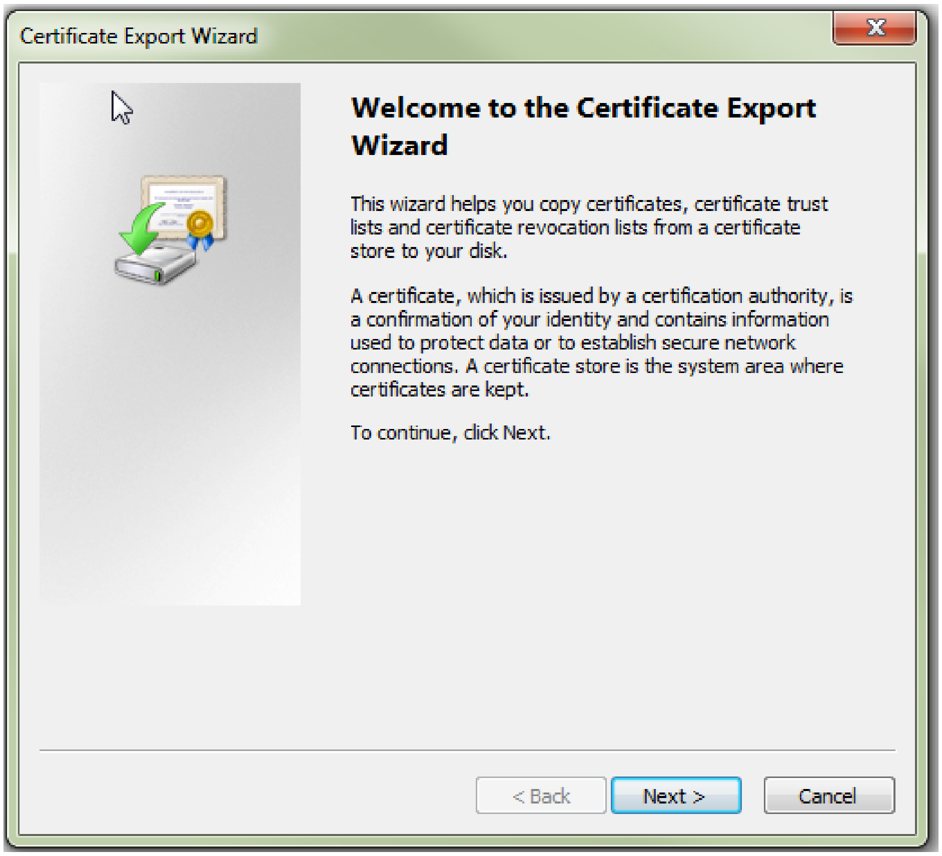
Show: <All>

Field	Value
Version	V3
Serial number	00 85 28 bb 80 f4 bf c1 7d 8e ...
Signature algorithm	sha256RSA
Signature hash algorithm	sha256
Issuer	Amazon Web Services O=Ama...
Valid from	31 August 2016 23:13:56
Valid to	01 January 2050 05:29:59
Subject	AWS IoT Certificate

Learn more about certificate details

3. In "Certificate Export Wizard" dialogue box, click "Next".

Figure 3-4. CERTIFICATE EXPORT WIZARD DIALOGUE BOX



Certificate Export Wizard

Welcome to the Certificate Export Wizard

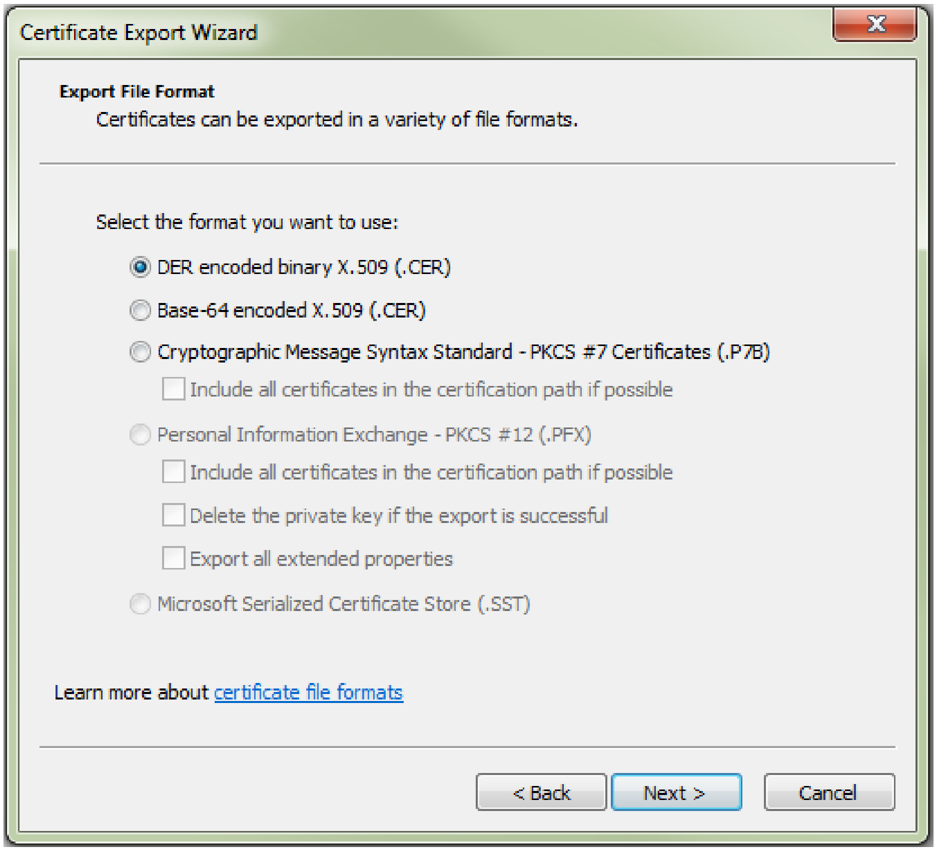
This wizard helps you copy certificates, certificate trust lists and certificate revocation lists from a certificate store to your disk.

A certificate, which is issued by a certification authority, is a confirmation of your identity and contains information used to protect data or to establish secure network connections. A certificate store is the system area where certificates are kept.

To continue, click Next.

4. Select the default option "DER encoded binary X.509 (.CER)" format to export and click "Next".

Figure 3-5. SELECT FORMAT IN CERTIFICATE EXPORT WIZARD



Certificate Export Wizard

Export File Format

Certificates can be exported in a variety of file formats.

Select the format you want to use:

● DER encoded binary X.509 (.CER)

○ Base-64 encoded X.509 (.CER)

○ Cryptographic Message Syntax Standard - PKCS #7 Certificates (.P7B)

□ Include all certificates in the certification path if possible

○ Personal Information Exchange - PKCS #12 (.PFX)

□ Include all certificates in the certification path if possible

□ Delete the private key if the export is successful

□ Export all extended properties

○ Microsoft Serialized Certificate Store (.SST)

Learn more about certificate file formats

5. Store the file in the name of "atmelwinc.cer".
6. Rename the AWS downloaded private key file as "atmelwinc.key".
7. Store both the "atmelwinc.cer" and "atmelwinc.key" files in the “***src/tls_cert_store/***” folder of Firmware Upgrade project (available with Firmware Upgrade project- not supplied with this package).
8. Flash the "WINC1500" firmware using the ***src/ download_all_sb_samw25_xplained_pro.bat*** for SAMW25 device or ***download_all_sb_samd21_xplained_pro.bat*** for SAMD21 device (available with firmware upgrade project- not supplied with this package), after loading the generated certificate and key file.

4. Configuring the Demo Application (WINC1500_AWS_RSA_EXAMPLE)

The **WINC1500_AWS_RSA_EXAMPLE** application publishes a message (on a certain topic) to the AWS Cloud MQTT Message Broker from the device, which is configured as **PUBLISHER**.
Note: The Publish event is triggered by a button press.

To view the published messages, the AWS MQTT Client is used to subscribe to the same topic on which the SAMW25/SAMD21 is publishing. The device which is configured as **SUBSCRIBER** receives the messages.

4.1 WINC1500_AWS_RSA_EXAMPLE Application Configuration

4.1.1 WLAN Configuration

In main.h, set the following configuration parameters according to the Wireless AP settings.

```
/** Wi-Fi Settings */

#define MAIN_WLAN_SSID "DEMO" /**< Destination SSID */

#define MAIN_WLAN_AUTH M2M_WIFI_SEC_WPA_PSK /**< Security manner */

#define MAIN_WLAN_PSK "123456" /**< Password for Destination SSID */
```

4.1.2 AWS IoT Settings

In aws_iot_config.h, set the following configuration parameters according to the AWS account

```
// Get from console

// =================================================

// To be Modified based on the user account

#define AWS_IOT_MQTT_HOST "XXXXXXXXXXXX.iot.us-west-2.amazonaws.com"

#define AWS_IOT_MQTT_PORT 8883

#define AWS_IOT_MQTT_CLIENT_ID "SAMD21_MQTT"

#define AWS_IOT_MY_THING_NAME "SAMD21_MQTT"

#define AWS_IOT_ROOT_CA_FILENAME " "

#define AWS_IOT_CERTIFICATE_FILENAME " "

#define AWS_IOT_PRIVATE_KEY_FILENAME " "
```

4.1.3 Application Device Settings

In main.c, set the following configuration parameters to set the device role either SUBSCRIBER or PUBLISHER, and the associated subscribe channel and publish channel to receive and send messages.

```
/*Role of the device*/

//#define SUBSCRIBER

#define PUBLISHER

#ifdef SUBSCRIBER

#define CLIENT_ID "WINC1500_Sub"
```

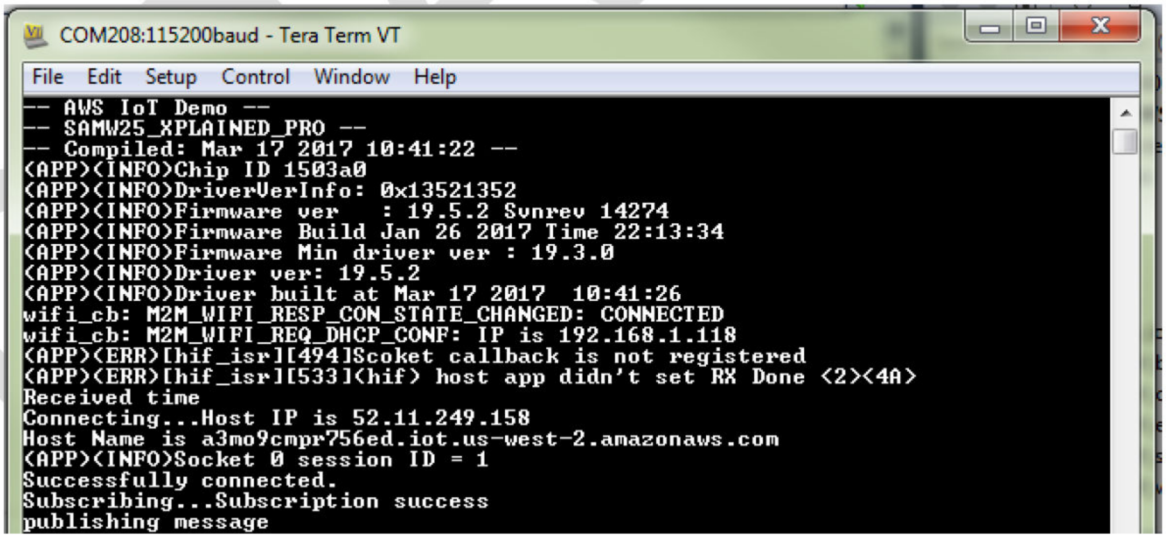
```
#define SUBSCRIBE_CHANNEL "WINC1500_IOT/sub"
#define PUBLISH_CHANNEL "WINC1500_IOT/pub"
#else
#define CLIENT_ID "WINC1500_Pub"
#define SUBSCRIBE_CHANNEL "WINC1500_IOT/pub"
#define PUBLISH_CHANNEL "WINC1500_IOT/sub"
#endif
```

5. Running the Demo

Perform the following steps to run the demo:

1. Configure the AWS IoT Account, refer to AWS IoT Account Setup.
2. Generate the Thing and Certificate from AWS IoT console.
3. Convert the Certificate to the .cer format and rename both the key and certificate, as mentioned in section Programming Certificates.
4. After loading the generated certificate and key file from AWS, flash the WINC1500 firmware using the ***src/ download_all_sb_samw25_xplained_pro.bat*** for SAMW25 device or ***download_all_sb_samd21_xplained_pro.bat*** for SAMD21 device (available with firmware upgrade project- not supplied with this package).
5. Configure the Application **WINC1500_AWS_RSA_EXAMPLE** , refer to **Configuring the Demo Application (WINC1500_AWS_RSA_EXAMPLE)**.
6. Build and run **WINC1500_AWS_RSA_EXAMPLE**.
7. Configure one device as `PUBLISHER` and another device as `SUBSCRIBER`.
8. Once the "Successfully connected" status is displayed on the serial console (115200 8N1 configuration). We can publish and receive the messages.

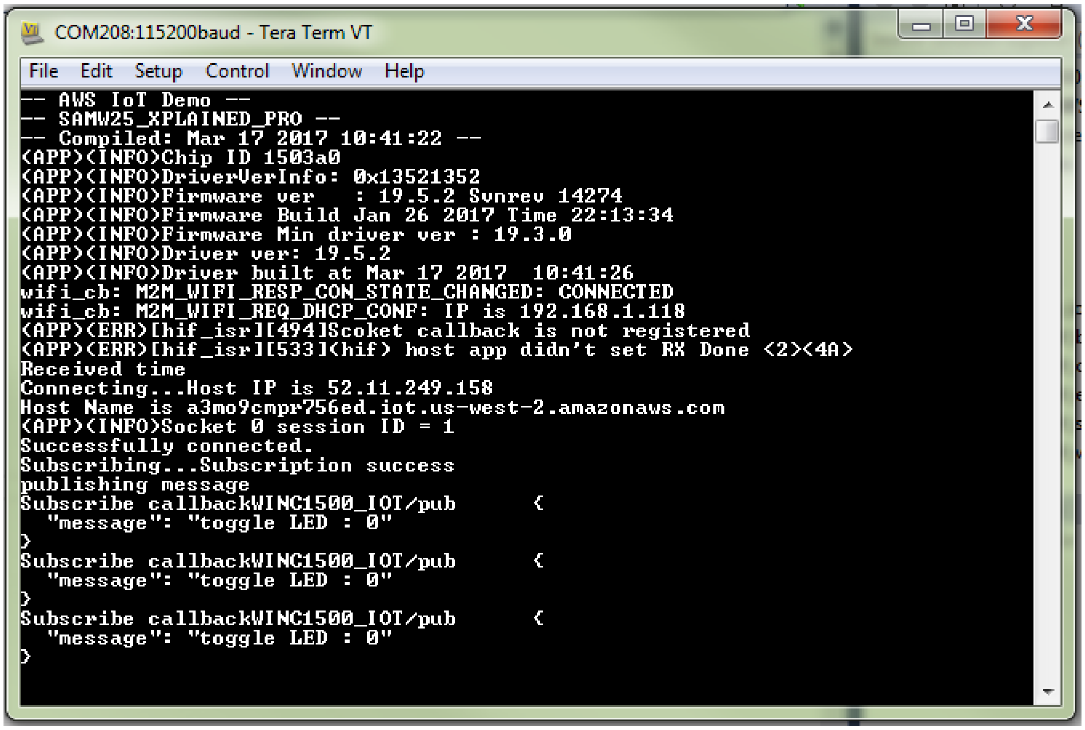
Figure 5-1. PUBLISHING MESSAGE ON TERA TERM VT WINDOW



```
-- AWS IoT Demo --
-- SAMW25_XPLAINED_PRO --
-- Compiled: Mar 17 2017 10:41:22 --
(APP)(INFO)Chip ID 1503a0
(APP)(INFO)DriverVerInfo: 0x13521352
(APP)(INFO)Firmware ver   : 19.5.2 Svnrev 14274
(APP)(INFO)Firmware Build Jan 26 2017 Time 22:13:34
(APP)(INFO)Firmware Min driver ver : 19.3.0
(APP)(INFO)Driver ver: 19.5.2
(APP)(INFO)Driver built at Mar 17 2017  10:41:26
wifi_cb: M2M_WIFI_RESP_CON_STATE_CHANGED: CONNECTED
wifi_cb: M2M_WIFI_REQ_DHCP_CONF: IP is 192.168.1.118
(APP)(ERR)[hif_isr][494]Scoket callback is not registered
(APP)(ERR)[hif_isr][533](hif) host app didn't set RX Done <2><4A>
Received time
Connecting...Host IP is 52.11.249.158
Host Name is a3mo9cmpr756ed.iot.us-west-2.amazonaws.com
(APP)(INFO)Socket 0 session ID = 1
Successfully connected.
Subscribing...Subscription success
publishing message
```

9. The device configured as `PUBLISHER` publishes the message on the Press button event and the same is received by the `SUBSCRIBER` device.
10. Press the button SW0 on `PUBLISHER device`. This publishes a message.
11. On the MQTT Client, the message is displayed on the console (if the topic has been subscribed).
12. The console log for the device is provided below for reference.

Figure 5-2. TERA TERM VT WINDOW


COM208:115200baud - Tera Term VT
File Edit Setup Control Window Help

```
-- AWS IoT Demo --
-- SAMW25_XPLAINED_PRO --
-- Compiled: Mar 17 2017 10:41:22 --
(APP)(INFO)Chip ID 1503a0
(APP)(INFO)DriverVerInfo: 0x13521352
(APP)(INFO)Firmware ver   : 19.5.2 Svnrev 14274
(APP)(INFO)Firmware Build Jan 26 2017 Time 22:13:34
(APP)(INFO)Firmware Min driver ver : 19.3.0
(APP)(INFO)Driver ver: 19.5.2
(APP)(INFO)Driver built at Mar 17 2017  10:41:26
wifi_cb: M2M_WIFI_RESP_CON_STATE_CHANGED: CONNECTED
wifi_cb: M2M_WIFI_REQ_DHCP_CONF: IP is 192.168.1.118
(APP)(ERR)[hif_isr][494]Scoket callback is not registered
(APP)(ERR)[hif_isr][533](hif) host app didn't set RX Done <2><4A>
Received time
Connecting...Host IP is 52.11.249.158
Host Name is a3mo9cmpr756ed.iot.us-west-2.amazonaws.com
(APP)(INFO)Socket 0 session ID = 1
Successfully connected.
Subscribing...Subscription success
publishing message
Subscribe callbackWINC1500_IOT/pub      {
  "message": "toggle LED : 0"
}
Subscribe callbackWINC1500_IOT/pub      {
  "message": "toggle LED : 0"
}
Subscribe callbackWINC1500_IOT/pub      {
  "message": "toggle LED : 0"
}
```

6. Document Revision History

Revision A (04/2017)

Section	Changes
Document	Initial Release.

The Microchip Web Site

Microchip provides online support via our web site at http://www.microchip.com/. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Customer Change Notification Service

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at http://www.microchip.com/. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

Product Identification System

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

Note:

1. Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.
2. Small form-factor packaging options may be available. Please check http://www.microchip.com/packaging for small-form factor package availability, or contact your local Sales Office.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip’s Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip’s code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

ISBN: 978-1-63277-605-1

Quality Management System Certified by DNV

ISO/TS 16949
Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/
support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon

Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-3326-8000
Fax: 86-21-3326-8021

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-3019-1500

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

France - Saint Cloud
Tel: 33-1-30-60-70-00

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820