

# User Guide

## SSK Cloud Connect

Date: November 23, 2020 | Version 1.0  
FINAL

---



The Solutions People



### Confidentiality Notice

Copyright (c) 2020 eInfochips. - All rights reserved

This document is authored by eInfochips and is eInfochips intellectual property, including the copyrights in all countries in the world. This document is provided under a license to use only with all other rights, including ownership rights, being retained by eInfochips. This file may not be distributed, copied, or reproduced in any manner, electronic or otherwise, without the express written consent of eInfochips.

# CONTENTS

<b>1</b>	<b>INTRODUCTION.....</b>	<b>3</b>
1.1	Purpose of the Document .....	3
1.2	Prerequisite .....	3
<b>2</b>	<b>I AM USER .....</b>	<b>4</b>
2.1	Description .....	4
2.2	Creating An I AM User .....	4
2.3	Listing I AM Users .....	5
<b>3</b>	<b>DEVICE.....</b>	<b>6</b>
3.1	Description .....	6
3.2	Add a Device .....	6
3.3	Listing Devices .....	6
3.3	Blacklisting Devices .....	7
3.3	Whitelisting Devices .....	8
<b>4</b>	<b>GREEN GRASS .....</b>	<b>9</b>
4.1	Description .....	9
4.2	Create A Greengrass Group .....	9
4.3	Listing Green grass Group .....	10
4.4	Green grass Subscription .....	10
4.5	Green grass Device.....	12
4.6	Green grass Deployment.....	13
<b>5</b>	<b>OTA.....</b>	<b>14</b>
5.1	Description .....	14
5.2	Create a FreeRTOS OTA update job (Schedule OTA).....	14
5.3	Create Custom Job (Schedule Job).....	15
5.3	Listing OTA.....	16
5.3	Create a Code Signing profile.....	17
<b>6</b>	<b>CERTIFICATE .....</b>	<b>18</b>
6.1	Description .....	18
6.2	Register CA .....	18
6.3	Listing Certificate .....	19
6.4	Add a certificate .....	20
6.5	Create a certificate(One click).....	21
<b>7</b>	<b>POLICY .....</b>	<b>22</b>
7.1	Description .....	22
7.2	Creating a Policy .....	22
7.3	Listing the Policy .....	23
7.4	Attach Policy .....	24
7.5	Detach Policy.....	24
<b>8</b>	<b>REFERENCES.....</b>	<b>6</b>

## 1 INTRODUCTION

### 1.1 Purpose of the Document

The SSK cloud connect guide provides an overview of the application to setup IAM User, Devices, AWS Greengrass, Certificate and Policies.

### 1.2 Prerequisite

This guide assumes that below configurations have been completed, as outlined in the SSK Cloud Connect Installation & Setup Guide:

- AWS Account
- AWS User Programmatic access
- AWS EC2 Instance service
- AWS RDS service
- AWS OTA Role Access
- User should be login in application by following below URL:

URL: <http://ec2-13-233-137-7.ap-south-1.compute.amazonaws.com/#/login>

#### Default details

User name: [seed.dev@infochips.com](mailto:seed.dev@infochips.com)

Password: 123456

Arrow Security Starter Kit

Home Contact

SSK  
Security Starter Kit

UserName:  
Enter email

Password:  
Password

Login

© 2020 Arrow Electronics, Inc. All rights reserved

Figure 1: Login page

**Note:** Aws cloud doesn't support any kind of field Email ID at user level, due to that the forgot password operation not supported.

## 2 IAM USER

### 2.1 Description

An AWS Identity and Access Management (IAM) user is an entity that you create in SSK Cloud connect to represent the person or application that uses it to interact with SSK Cloud connect. A user in SSK Cloud connect consists of a name and credentials.

### 2.2 Creating IAM User

An IAM user is a resource in IAM that has associated credentials and permissions. An IAM user can represent a person or an application that uses its credentials to make SSK Cloud connect requests.

The screenshot shows the 'Add User' form in the SSK Cloud Connect interface. The form is titled 'User / Add User' and includes the following fields and options:

- First Name:** Text input field with the value 'Abc'.
- Last Name:** Text input field with the value 'Xyz'.
- User Name:** Text input field with the value 'abc@xyz.com'.
- Email Id:** Text input field with the value 'abc@xyz.com'.
- Password:** Password input field with 16 dots.
- Mobile Number:** Text input field with the value '+1 999-999-9999'.
- Permission:** Checkboxes for 'User' (unchecked), 'Tech Support' (checked), and 'Admin' (checked).
- Group Name:** Dropdown menu with the value 'SEED\_DEV\_TEAM'.

At the bottom of the form are 'Cancel' and 'Add' buttons. The footer of the page reads '© 2020 Arrow Electronics, Inc. All rights reserved'.

Figure 2: Add IAM User

An IAM user can be described by the following:

- 1. First Name**  
The first name of the user with min 4 chars.
- 2. Last Name**  
The last name of the user.
- 3. User Name**  
The unique name of the user with min 4 chars. Using that user can login into system. It should be unique across AWS account.
- 4. Password**  
The password of the user. At least 16 chars with combination of one Upper case, Lower case, Digit and Special character.

## 5. Email Id

The Email Id of the user. It is not unique.

## 6. Mobile Number

The Mobile number of the user with min 8 chars.

## 7. Permission

- a. **User** - can access only dashboard access and demo page
- b. **Tech Support** - can manage device, group, OTA, certificate, policy
- c. **Admin** - can manage all functionality.

## 8. Group Name

Upon selection of group name user will be added in AWS Group to manage access

## 2.3 Listing IAM User

Once you create An IAM user, you can list down an IAM user at List User page. Which is shown Here in below images.

The screenshot displays the 'Security Starter Kit' web application interface. The top navigation bar includes 'Home', 'Contact', and 'Demo'. The sidebar on the left lists various management options: 'User' (highlighted), 'Device', 'Deny Device', 'Allow Device', 'Group', 'OTA Updates', 'Certificate', and 'Policy'. The main content area is titled 'User / List User' and features an 'Add User' button. Below this is a table listing existing users.

First Name	Last Name	User Name	Created Time
Arjun	Salariya	XXXXXXXXXXXXXXXXXX	28/08/2020 07:43
Darshak	Patel	XXXXXXXXXXXXXXXXXX	06/05/2020 10:26
		XXXXXXXXXX	22/04/2020 09:14
		XXXXXXXXXXXX	31/07/2020 08:12
		XXXXX	24/08/2020 11:32

At the bottom of the page, a copyright notice states: © 2020 Arrow Electronics, Inc. All rights reserved.

Figure 2: Listing IAM User

## 3 DEVICE

### 3.1 Description

AWS IoT provides a device registry that helps you manage your devices. A device is the representation of a device or logical entity. It can be a physical device or sensor (for example, a light bulb or a switch on a wall). It can also be a logical entity like an instance of an application or physical entity that does not connect to AWS IoT, but is related to devices that do (for example, a car that has engine sensors or a control panel).

Devices are identified by a name. Devices can also have attributes, which are name-value pairs you can use to store information about the device, such as its serial number or manufacturer. Adding your devices to the device registry allows you to manage and search for them more easily.

### 3.2 Add a Device

A device is the representation of a device or logical entity in the cloud. A device can be described by the following:

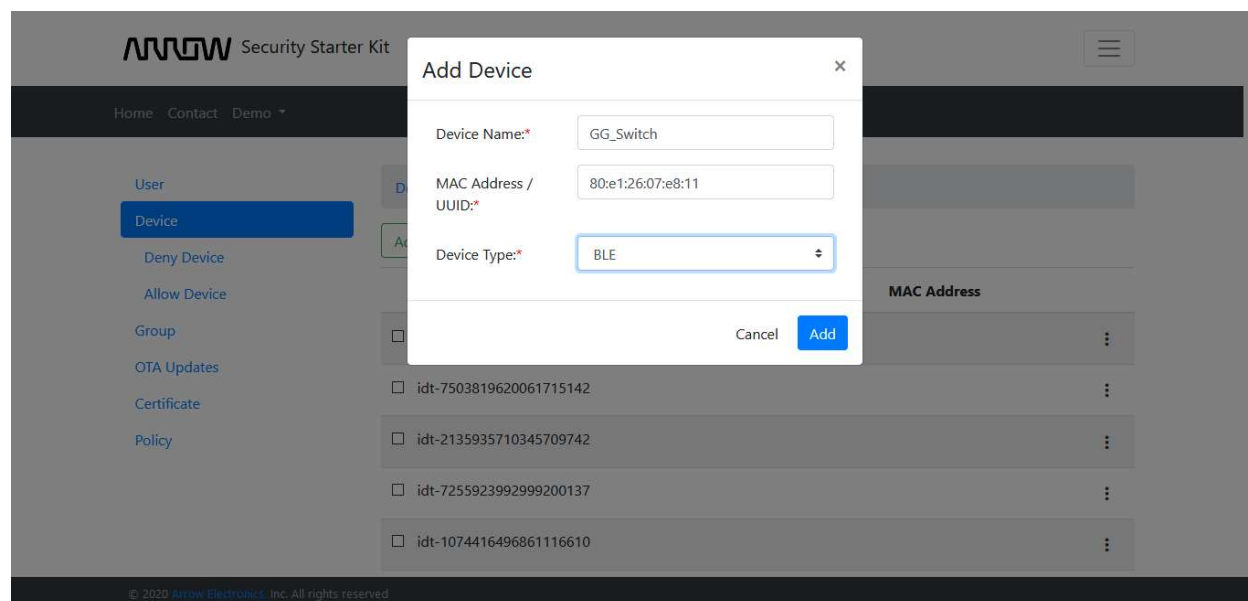


Figure 1: Add a device

1. **Device Name**  
The name of the device.
2. **MAC Address**  
The MAC Address of the device.

### 3. Device Type

Device types allow you to store description and configuration information that is common to all devices associated with the same device type. i.e. BLE, LTE-M, WIFI

### 3.3 Listing Devices

Listing of Devices can be seen in following figure with path SSK -> Device.

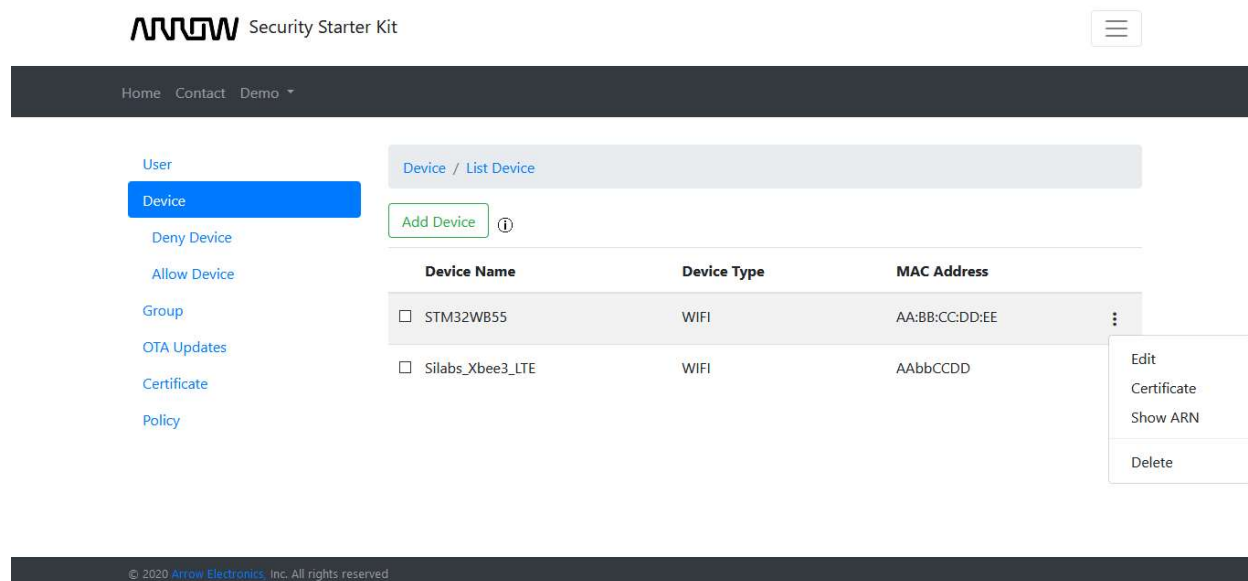


Figure 2: Listing devices

### 3.4 Deny Device

A Device which don't have at least one active certificate or added in Deny by MAC Address/UUID.

#### 1. Deny by Certificate

You can remove device from Deny list by attaching certificate.

#### 2. Deny by MAC Address

You can remove device from Deny list by clicking remove button.

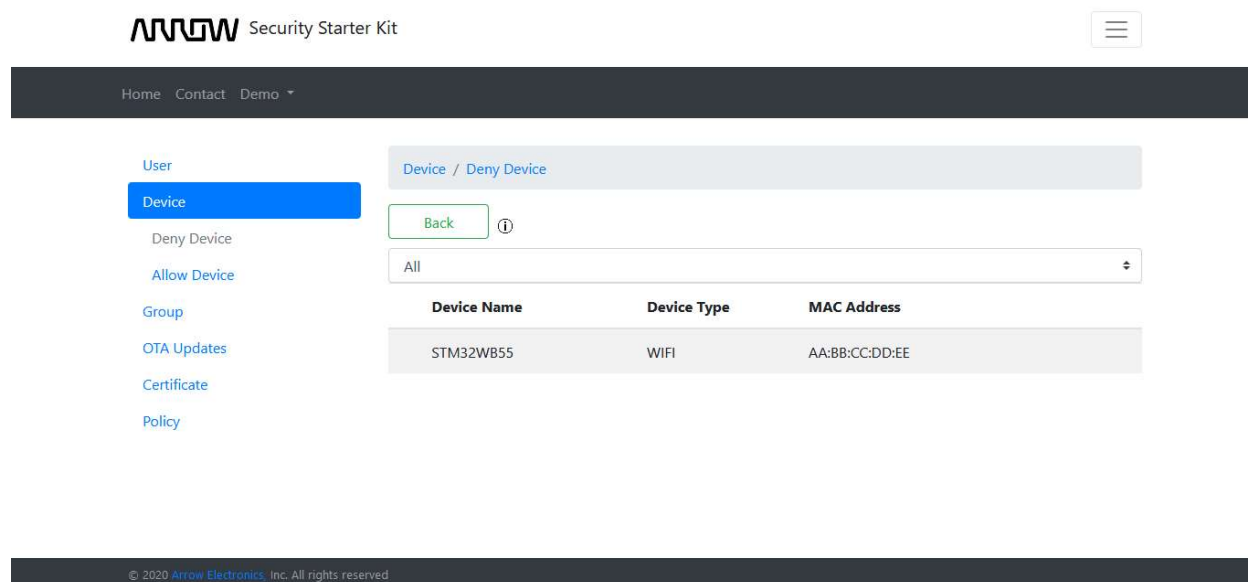


Figure 3: Listing blacklist devices

### 3.5 Allow Device

A Device which have at least one active certificate and not added in Deny by MAC Address/UUID.

#### 1. Deny by Certificate

You can add device to deny list by clicking button Deny By Certificate.

#### 2. Deny by MAC Address

You can add device to deny list by clicking button Deny By MAC Address/UUID.

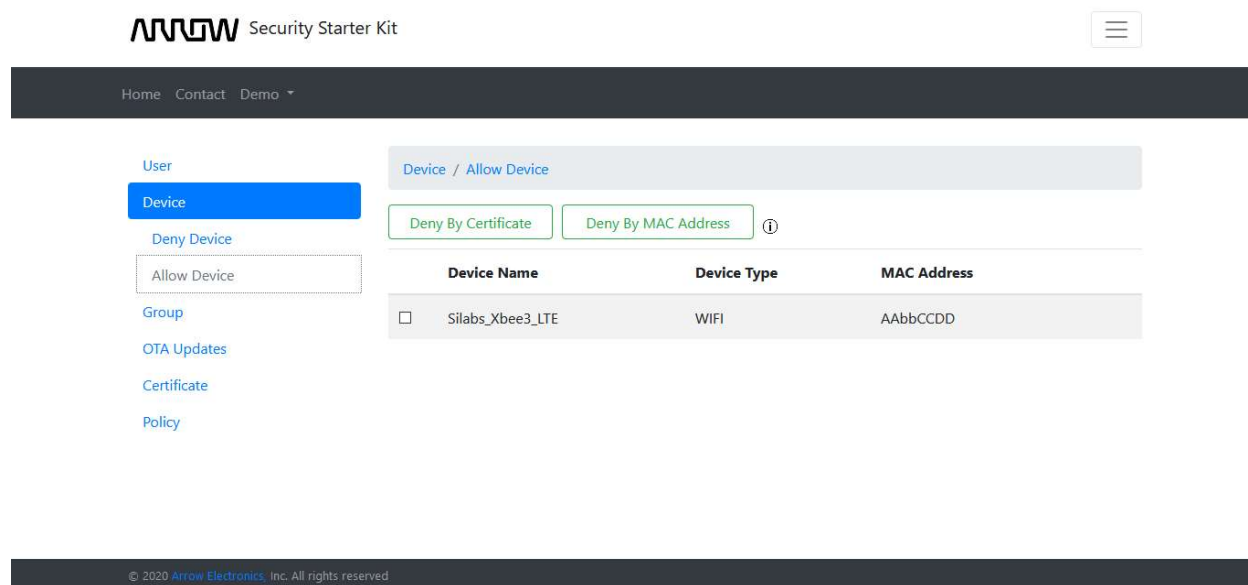


Figure 4: Listing whitelist devices



## 4 GREENGRASS

### 4.1 Description

An AWS IoT Greengrass lets your devices process the data they generate locally, while still taking advantage of AWS services when an internet connection is available.

### 4.2 Creating a Green grass Group

Setting up your Group requires you to provision a Core device in the IoT Registry, acquire a certificate for your Core, and assign an IAM role to your Group. A Group can be described by the following:

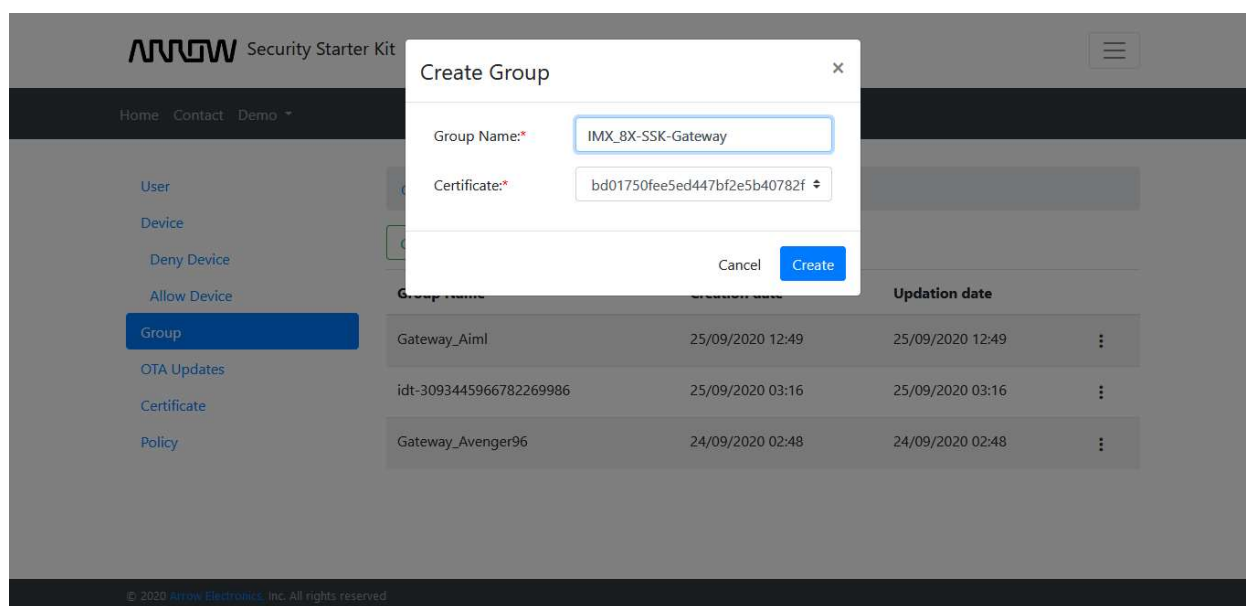


Figure 1: Create a Green grass Group

1. **Group Name**  
The name of the Green grass Group
2. **Certificate Id**  
The Certificate ID of the Green grass Core.

### 4.3 Listing Green grass Group

Listing of Greengrass group can be seen in following figure.

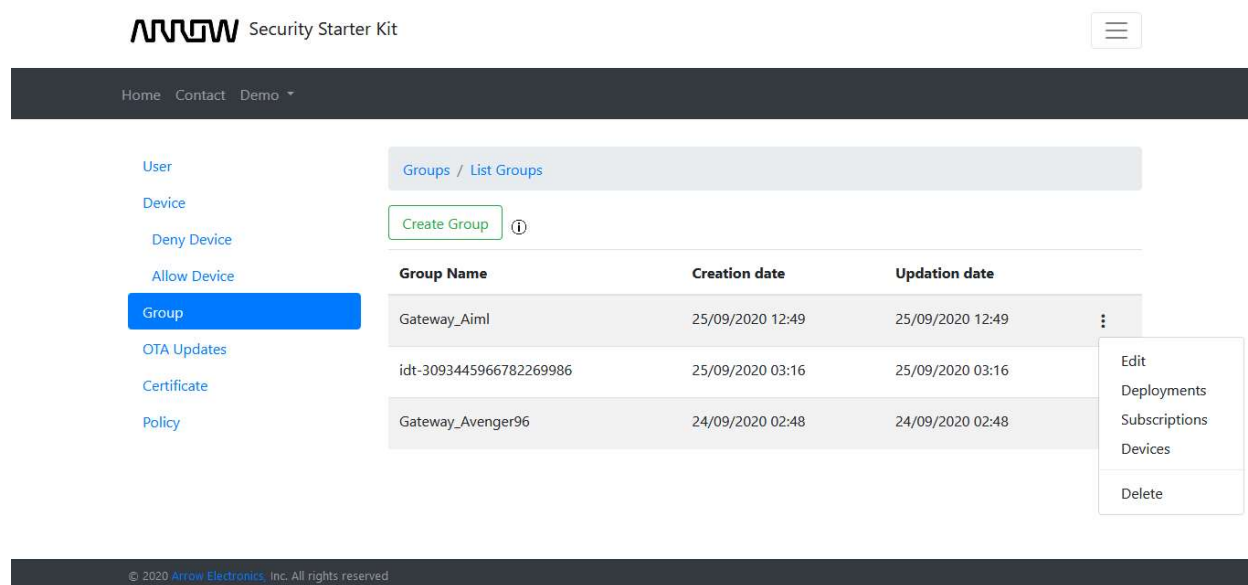


Figure 2: Listing Green grass Groups

### 4.4 Green grass Subscription

A Subscription consists of a source, target, and topic. The source is the originator of the message. The target is the destination of the message. The first step is selecting your source and target.

#### 4.4.1 Creating a Subscription

A Subscription can be added to group by the following figure:

1. **Source**

The name of the Source. Source can be Services like IoT Cloud, Local Shadow Service or Green grass devices.

2. **Topic**

The name of the Topic. AWS Cloud and device can communicate on given topic over MQTT.

3. **Target**

The name of the Target. Target can be Services like IoT Cloud, Local Shadow Service or Green grass devices.

Security Starter Kit

Home Contact Demo ▾

User

Device

Deny Device

Allow Device

**Group**

OTA Updates

Certificate

Policy

Groups / Add Subscription

Source: Local Shadow Service

Topic: \$aws/things/GG\_TrafficLight/shadow/update/rejected

Target: IOT Cloud

Cancel Add

© 2020 Arrow Electronics, Inc. All rights reserved

Figure 3: Create subscription To Group

#### 4.4.2 Listing Subscriptions

A Subscription List can be seen in following figure:

Security Starter Kit

Home Contact Demo ▾

User

Device

Deny Device

Allow Device

**Group**

OTA Updates

Certificate

Policy

Groups / List Subscription

Back Add Subscription ⓘ

	Source	Target	Topic	
1	Local Shadow Service	GG_Switch_AI	\$aws/things/GG_TrafficLight_AI/shadow/update/accepted	⋮
2	Local Shadow Service	GG_TrafficLight_AI	\$aws/things/GG_TrafficLight_AI/shadow/update/delta	⋮
3	Local Shadow Service	GG_TrafficLight_AI	\$aws/things/GG_TrafficLight_AI/shadow/update/rejected	⋮

© 2020 Arrow Electronics, Inc. All rights reserved

Figure 4: Listing subscriptions

## 4.5 Green grass Device

Greengrass Devices can be created by re-purposing an existing IoT Thing from your Registry or by creating new Registry items, and then adding them to a Greengrass Group.

### 4.5.1 Add a device

A device can be added to group by the following figure:

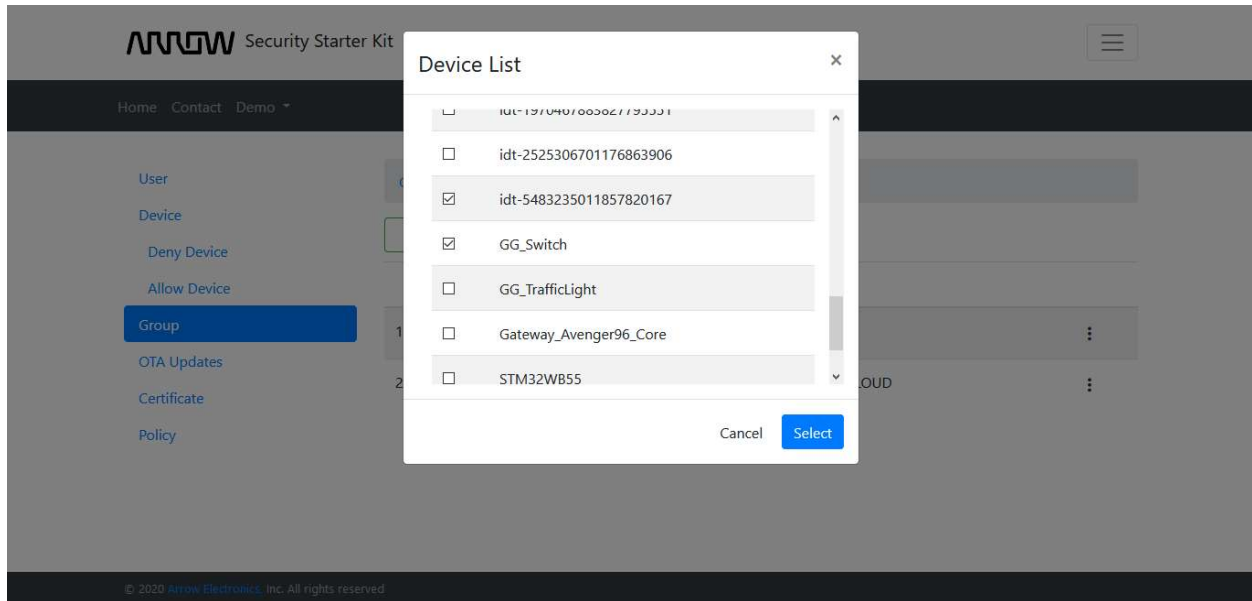


Figure 5: Add a device To Group

#### 1. Device Name

The name of the device.

## 4.5.2 Listing Devices

A Device List can be seen in following figure:

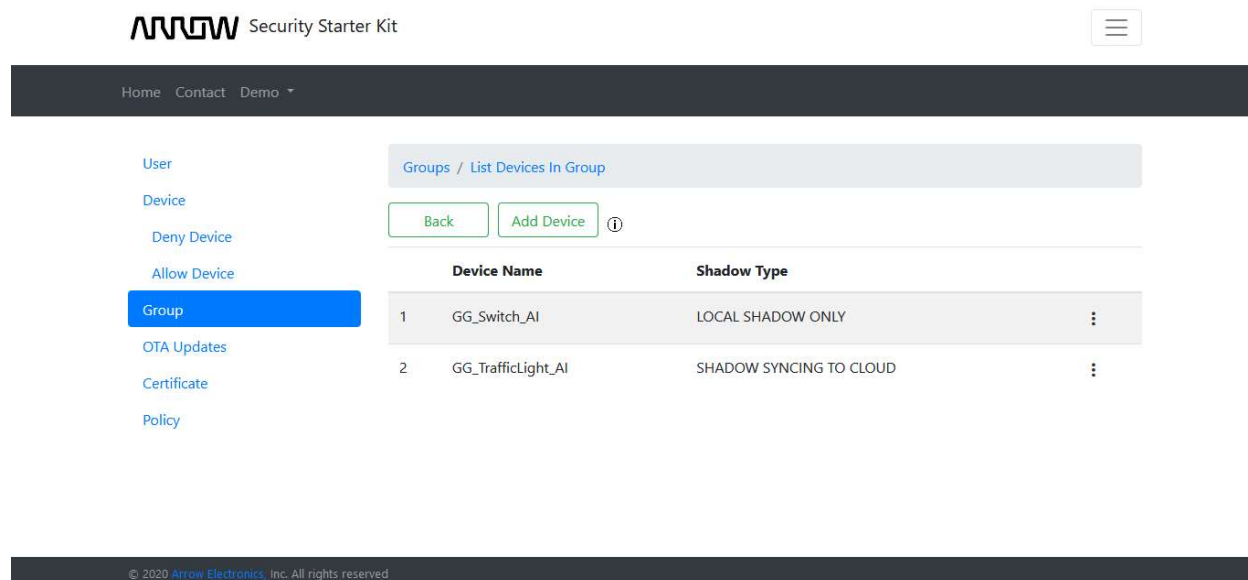


Figure 6: Listing devices

## 4.6 Green grass Deployment

A deployment of green grass group & core to device can be setup by following figure:

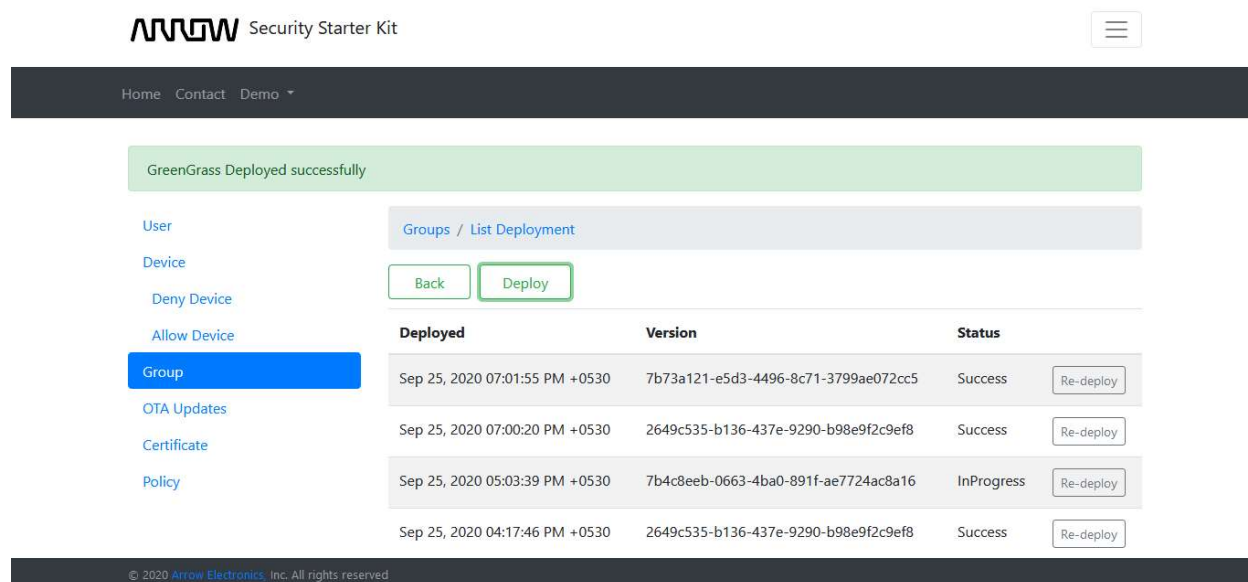


Figure 7: Deployment Greengrass

## 5 OTA UPDATES

### 5.1 Description

AWS IoT Device Management job orchestration and notification service allows you to define a set of remote operations called jobs that are sent to and executed on one or more devices connected to AWS IoT.

### 5.2 Create a FreeRTOS OTA update job (Schedule OTA)

This Over-the-air (OTA) update job will send your firmware image securely over MQTT or HTTP to FreeRTOS-based devices.

The screenshot shows the 'Schedule OTA' form in the AWS IoT Device Management console. The left sidebar contains navigation links: User, Device, Deny Device, Allow Device, Group, OTA Updates (highlighted), Certificate, and Policy. The main form area has the following fields:

- OTA Update ID:
- OTA Update Protocol:
- OTA Update Target:
- OTA Target Selection:
- OTA File Name:
- Device Firmware Path:
- Code Signing Profile:
- OTA Role ARN:

At the bottom of the form are 'Cancel' and 'Schedule' buttons. The footer of the console shows '© 2020 Arrow Electronics, Inc. All rights reserved.'

Figure 1: Schedule OTA

- OTA Update ID**  
A unique OTA update ID.
- OTA Update Protocol**  
The protocol that you choose must be supported by your device. If you select a protocol that is not supported by your device, the firmware update will be unsuccessful.
- OTA Update Target**  
Select the devices you want to include in this job.
- OTA Target Selection**  
Snapshot job is sent to all targets that when you create the job. After those targets complete the job (or report that they are unable to do so), the job is complete.
- OTA File Name**  
Name of firmware.

### 6. Device Firmware Path

This is the location and name to use when storing the firmware on the FreeRTOS device during OTA update. It is an optional field since certain devices do not store the image to a filesystem and may instead write directly to internal flash memory.

### 7. Code Signing Profile

Code signing ensures that devices only run code published by trusted authors and that the code has not been altered or corrupted since it was signed. You have three options for code signing.

### 8. OTA Role ARN

A Role which grants AWS IoT access to the S3, AWS IoT jobs and AWS Code signing resources to create an OTA update job.

## 5.3 Create Custom Job (Schedule Job)

Send a request to acquire an executable job file from one of your S3 buckets to one or more devices connected to AWS IoT.

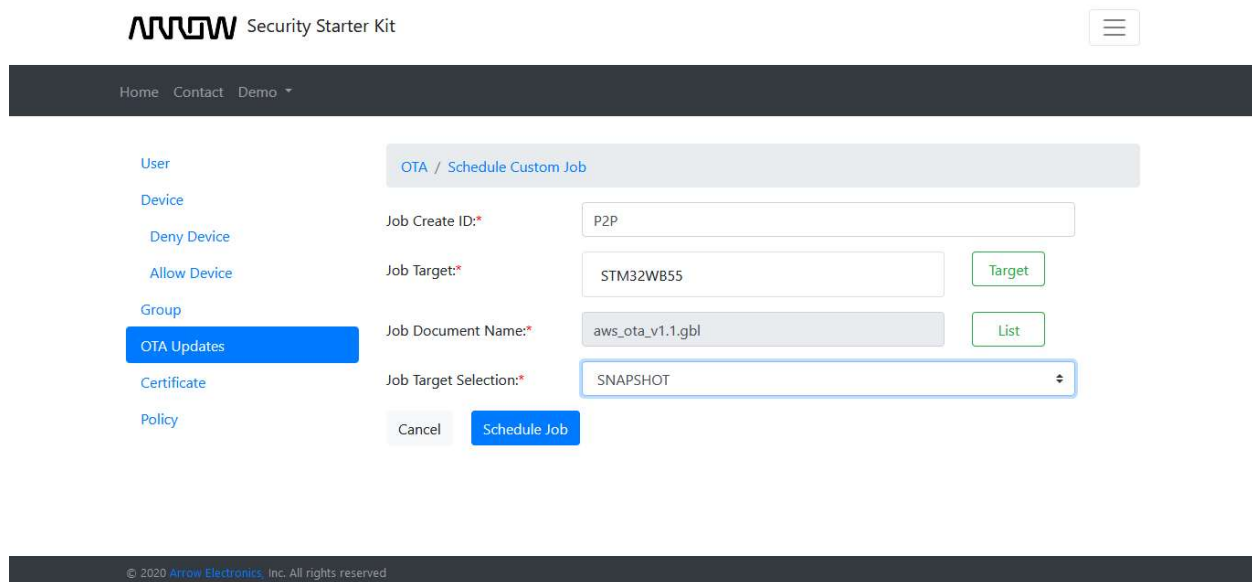


Figure 2: Schedule custom job

#### 1. Job Create ID

A unique job Id.

#### 2. Job Target

Select the devices you want to include in this job.

#### 3. Job Document Name

Upload a job file that defines what your job should do.

Job documents are JSON documents and should contain any information your devices need to perform a job. For example, a job document can contain one or more URLs where the device can download an update or some other data.

#### 4. Job Target Selection

Snapshot job is sent to all targets that when you create the job. After those targets complete the job (or report that they are unable to do so), the job is complete.

## 5.4 Listing OTA UPDATES

Listing of ota job can be seen in following figure:

The screenshot shows a web interface for scheduling an OTA update. On the left is a sidebar menu with options: User, Device, Deny Device, Allow Device, Group, OTA Updates (highlighted in blue), Certificate, and Policy. The main content area is titled 'OTA Updates / Schedule OTA'. It contains several form fields: 'OTA Update ID:\*' with the value 'ota\_stm32wb55'; 'OTA Update Protocol:\*' with a dropdown menu showing 'MQTT'; 'OTA Update Target:\*' with the value 'STM32WB55' and a green 'Target' button; 'OTA Target Selection:\*' with a dropdown menu showing 'SNAPSHOT'; 'OTA File Name:\*' with the value 'aws\_demos.sfb' and a green 'List' button; 'Device Firmware Path:\*' with the value 'firmware.bin'; 'Code Signing Profile:\*' with a dropdown menu showing 'SEED\_codesign\_profile'; and 'OTA Role ARN:\*' with a dropdown menu showing 'lamOTAUpdateRole'. At the bottom of the form are 'Cancel' and 'Schedule' buttons. The footer of the page reads '© 2020 Arrow Electronics, Inc. All rights reserved'.

Figure 3: Listing OTA Job



## 5.5 Create a Code Signing profile

The code signing profile contains information needed to create a code signing job. It specifies your device's hardware platform, certificate from AWS Certificate Manager, and the location of your code signing certificate path on your device.

**Create Code Signing Profile**

Profile Name:\* SEED\_codesign\_profile

Platform ID:\* Amazon FreeRTOS SHA256-ECDSA

Code Signing Certificate:\* Browse... ecdsasigner.crt

Certificate Private Key:\* Browse... ecdsasigner.key

Path Name:\* /home/certificate/ecdsasigner.crt

Cancel Create

Profile Name	Platform ID	Status
4 VireshCodeSign2	AmazonFreeRTOS-Default	Active
5 wb55_16july_profile	AmazonFreeRTOS-Default	Active

© 2020 Amazon Web Services, Inc. All rights reserved.

Figure 4: Create code signing profile

1. **Profile Name**  
A unique profile name.
2. **Platform ID**  
Select the platform.
3. **Code Signing Certificate**  
Upload signing certificate
4. **Certificate Private Key**  
Upload private key
5. **Path Name**  
Select the path name of certificate on device

## 6 CERTIFICATE

### 6.1 Description

A certificate is used to authenticate your device's connection to AWS IoT.

### 6.2 Register CA

To use your own X.509 certificates, you must register a CA certificate with AWS IoT. The CA certificate can then be used to sign device certificates. You can register up to ten CA certificates with the same subject field and public key per AWS account. This allows you to have more than one CA sign your device certificates.

- Step 1: Generate a key pair for the private key verification certificate
- Step 2: Copy this registration code (From Setting page at top-right corner)
- Step 3: Create a CSR with this registration code Put the registration code in the Common Name field
- Step 4: Use the CSR that was signed with the CA private key to create a private key verification certificate
- Step 5: Upload the CA certificate (rootCA.pem)
- Step 6: Upload the verification certificate (verificationCert.crt) :

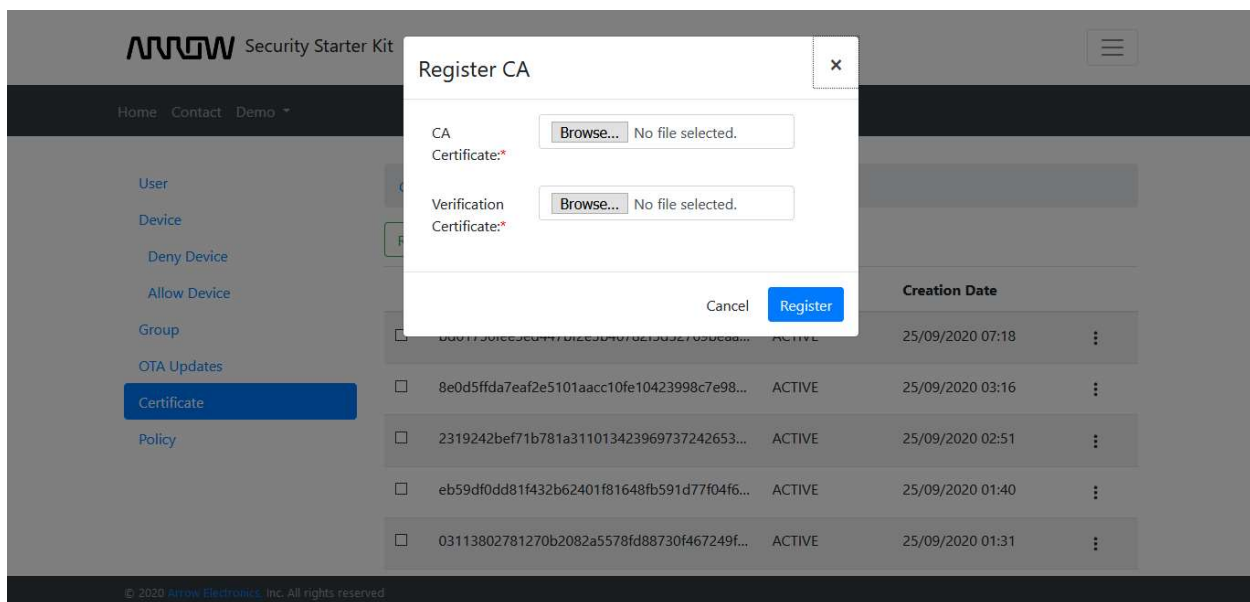


Figure 1: Register CA

1. **Root CA File**  
Select & upload root CA file.
2. **Verification Certificate File**  
Select & upload verification certificate file.

## 6.3 Listing Certificates

Listing of register of all certificate can be seen in following figure:

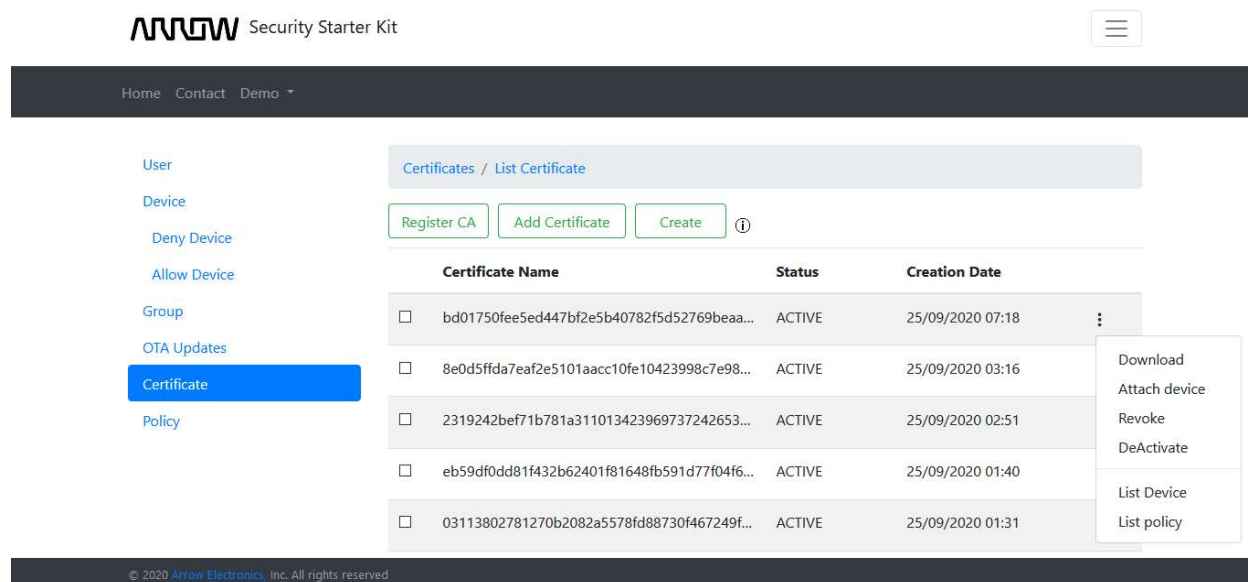


Figure 2: Listing Certificates

Certificate will be having following operations:

1. **Download**  
Download certificate file in PEM format.
2. **Attach device**  
Attach certificate to device.

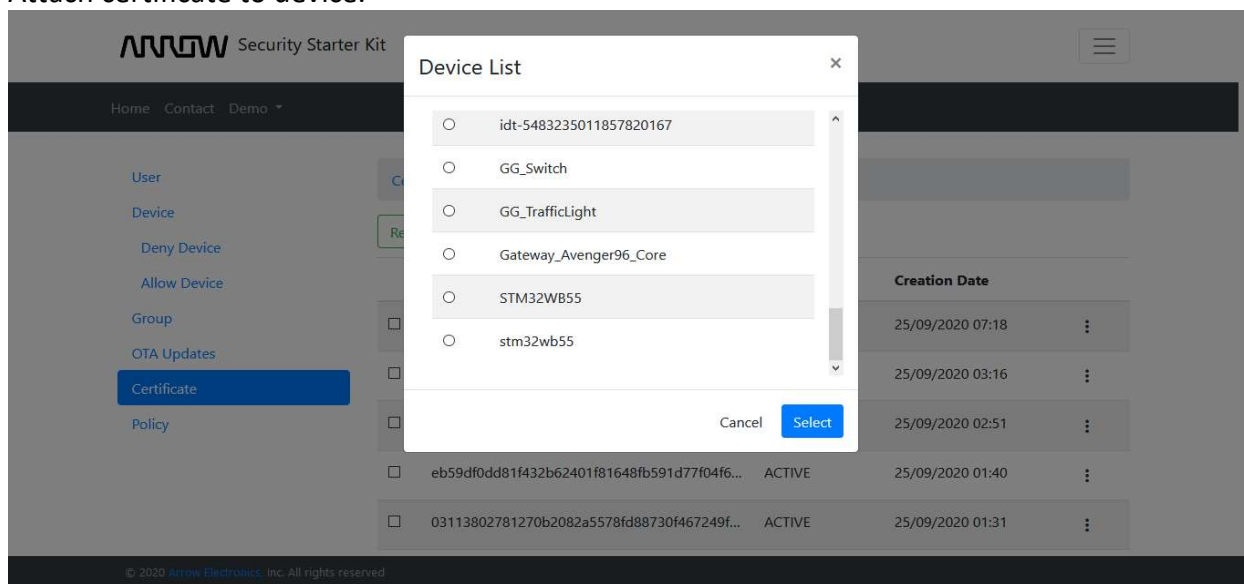


Figure 3: Attach certificate to Device

### 3. Revoke

To Revoke certificate, once certificate will be revoked then it will not be activate again. Only you can delete certificate.

### 4. Deactivate

To Change status of certificate to inactive, once revoked certificate then deactivate option will not be available.

### 5. Delete

To Delete certificate, if certificate exists other active then this option will be available.

### 6. List Device

List down all devices attached to certificate.

### 7. List Policy

List down all policy attached to certificate.

## 6.4 Add a Certificate

Select or register the CA certificate used to sign your device certificates. To use device certificates that are not signed by a registered CA, just select next.

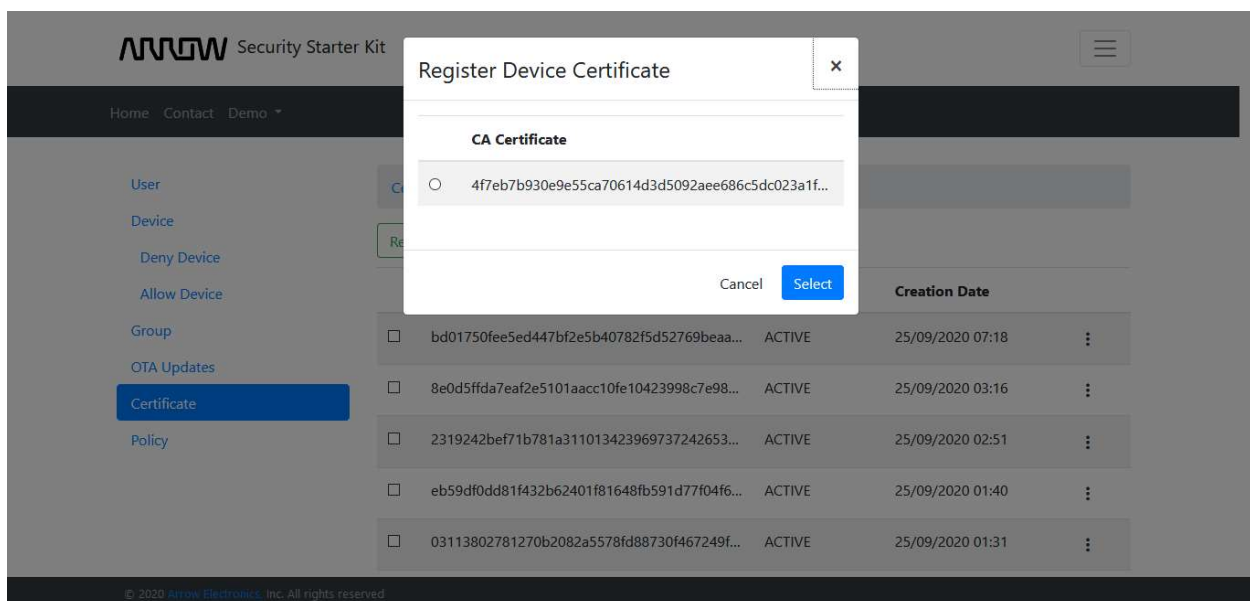


Figure 4: Add Certificate Part 1

Paste the existing certificate PEM file.

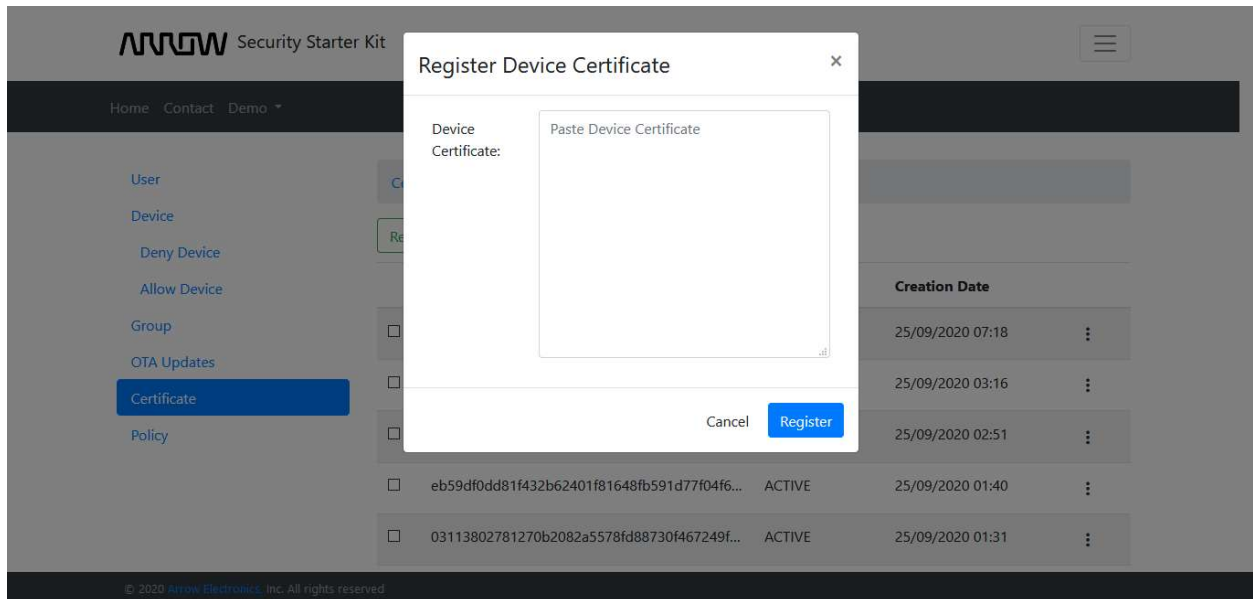


Figure 5: Add Certificate part 2

1. **Registered CA**  
Select CA ID from list of CA.
2. **Existing Certificate File**  
Paste the Certificate PEM file.

## 6.5 Create a Certificate (One Click)

This will generate a certificate, public key, and private key using AWS IoT's certificate authority.

## 7 POLICY

### 7.1 Description

AWS IoT policies grant or deny access to AWS IoT resources such as things, thing shadows, and MQTT topics. A device or user can invoke AWS IoT operations only if they are granted the appropriate permissions.

Policies give permissions to AWS IoT clients regardless of the authentication mechanism they use to connect to AWS IoT. To control which resources a device can access, attach one or more AWS IoT policies to the certificate associated with the device.

### 7.2 Creating 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).

**Security Starter Kit**

Home Contact Demo ▾

User  
Device  
Deny Device  
Allow Device  
Group  
OTA Updates  
Certificate  
**Policy**

Policy / Edit Policy

Policy Name: STM32WB55-policy

Policy Document:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*"
    }
  ]
}
```

Cancel Update

© 2020 Arrow Electronics, Inc. All rights reserved

Figure 1: Create Policy

- 1. Policy Name**  
The name of the policy.
- 2. Policy Document**  
Policy document define the types of actions that can be performed by a resource. It will not allow any spaces

Policy document Description can be seen below:

Policy document has mainly three elements

- i. Effect – Allow/ Deny
- ii. Action
 

Action can be define by following type.

  - a. `iot:*`
  - b. `iot:Publish`
  - c. `iot:Subscribe`
  - d. `iot:Connect`
  - e. `iot:Receive`
  - f. `iot:UpdateThingShadow`
  - g. `iot:GetThingShadow`
  - h. `iot>DeleteThingShadow`
- iii. Resource ARN
 

Resource could be client ID ARN, topic ARN or topic filter ARN

### 7.3 Listing the Policy

Listing of policy can be seen in following figure:

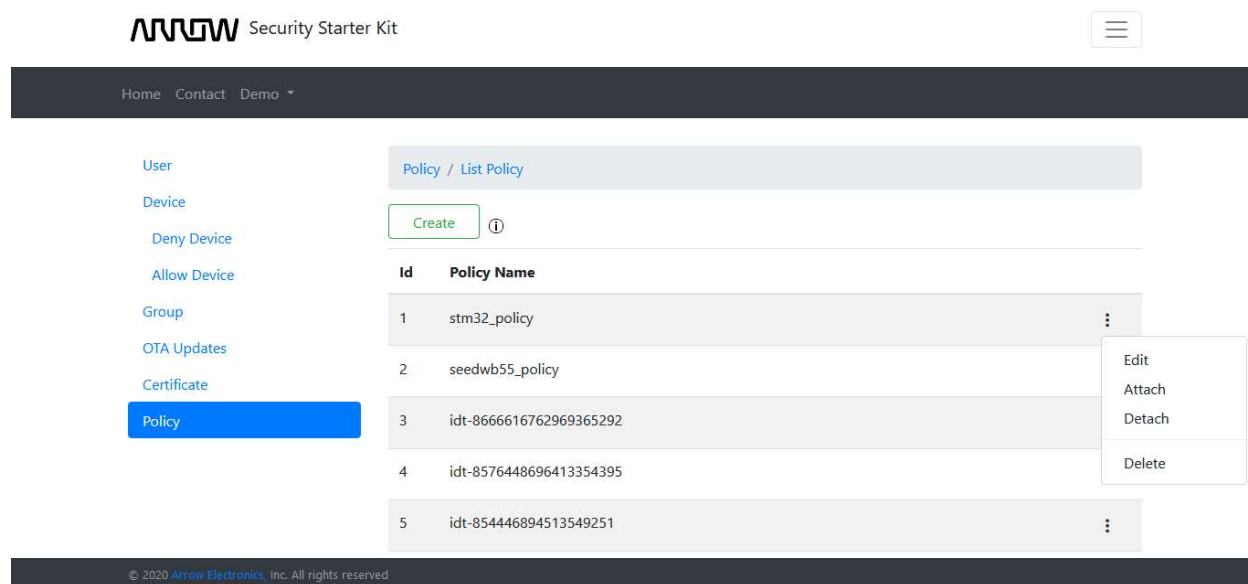


Figure 2: Listing Green grass Groups

## 7.4 Attach Policy

Policy can be attach to resource like Certificate, cognito which can be seen here in following figure

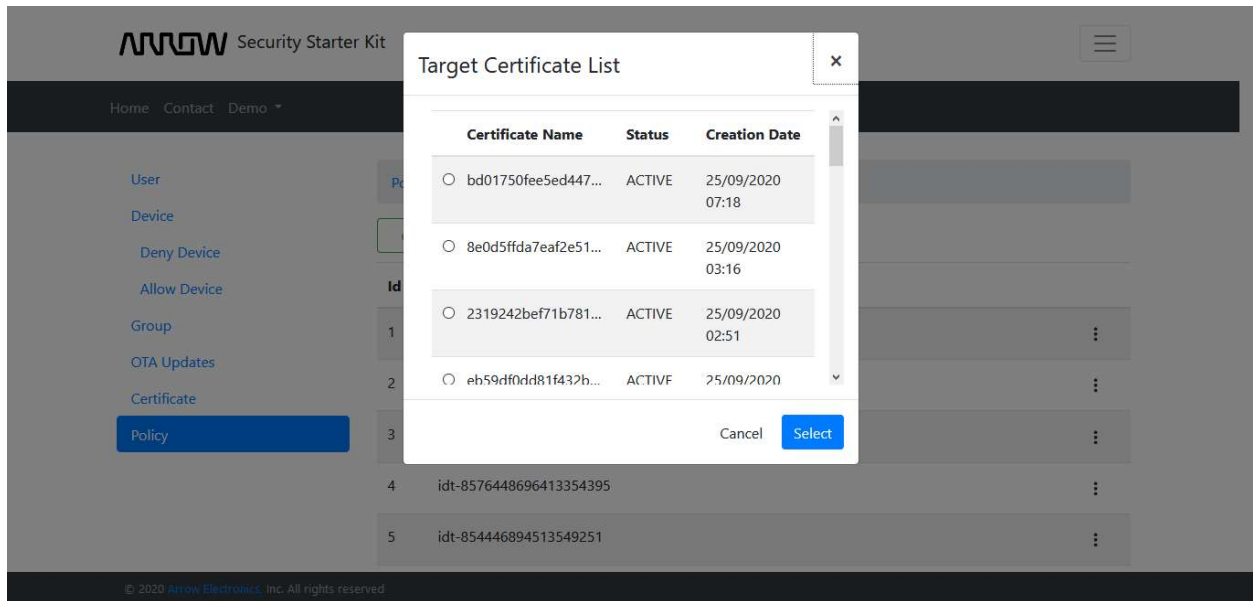


Figure 3: Attach policy to Certificate

## 7.5 Detach Policy

Detach policy from resource like Certificate, cognito can be seen here in following figure:

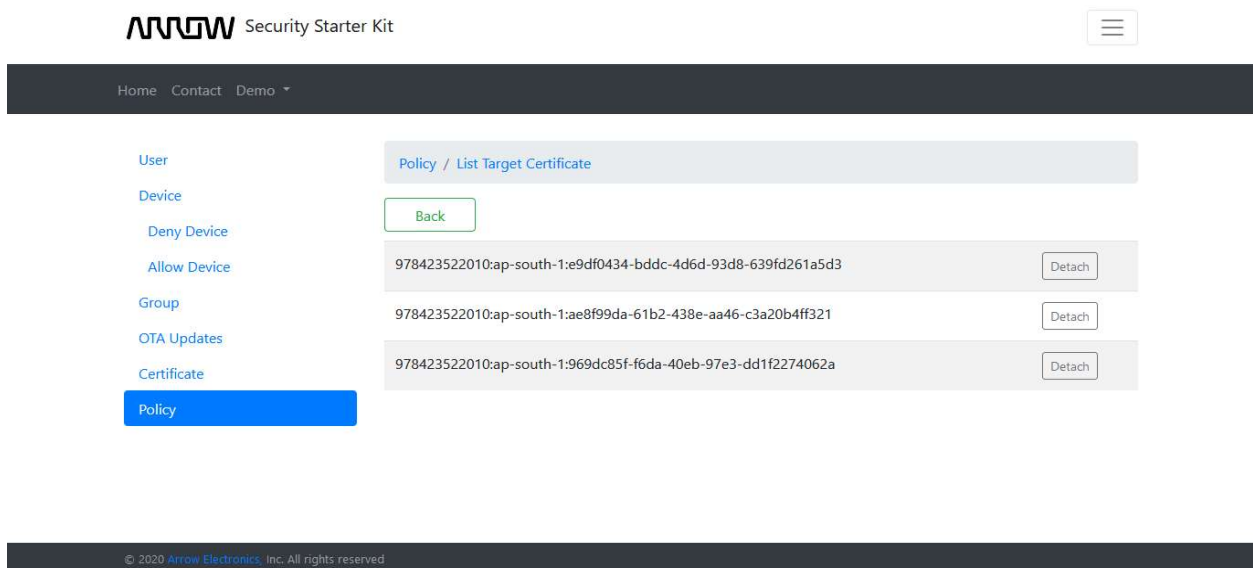


Figure 4: Detach policy from resource



## 8 REFERENCES

- [1] <https://docs.aws.amazon.com/greengrass/latest/developerguide/gg-dg.pdf>
- [2] <https://aws.amazon.com/blogs/iot/using-a-trusted-platform-module-for-endpoint-device-security-in-aws-iot-greengrass/>
- [3] <https://docs.aws.amazon.com/iot/latest/developerguide/register-CA-cert.html>
- [4] <https://aws.amazon.com/console/>