ARM® Cortex®-M

32-bit Microcontroller

Nuvoton-AWS-IoT
User Manual

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

Table of Contents

1 OVERVIEW

This user manual is aimed to give users a fast introduction to how to use NuMaker-PFM-M487 board to connect to the AWS IoT.

1.1 Introduction to AWS IoT

AWS IoT is a managed cloud platform that lets connected devices easily and securely interact with cloud applications and other devices. AWS IoT can support billions of devices and trillions of messages, and can process and route those messages to AWS endpoints and to other devices reliably and securely. Visit AWS IoT website: https://aws.amazon.com/ for more information.

1.2 Introduction to NuMaker-PFM-M487 Board

The NuMaker-PFM-M487 is a development board based on an ARM® Cortex®-M4 microcontroller (MCU) – M480 series which has very rich peripherals.

This board is provided by Nuvoton and created specially to support the ARM® mbed™ IoT Device Platform, and let user easily to develop the IoT application program on this board. The NuMaker-PFM-M487 also provides user many useful and powerful learning materials for how to develop and verify the application programs through the peripherals and interfaces on MCU and this board.

Furthermore, this board also provides an Arduino UNO compatible interface for user to develop the specific function with any of Arduino modules or kits. Regarding to the Arduino, user can link directly to the Wikipedia website: en.wikipedia.org/wiki/Arduino to get more detailed introductions.

The NuMaker-PFM-M487 board consists of M487 Platform and Nu-Link-Me ICE Bridge. Figure 1-1 shows the NuMaker-PFM-M487 board.

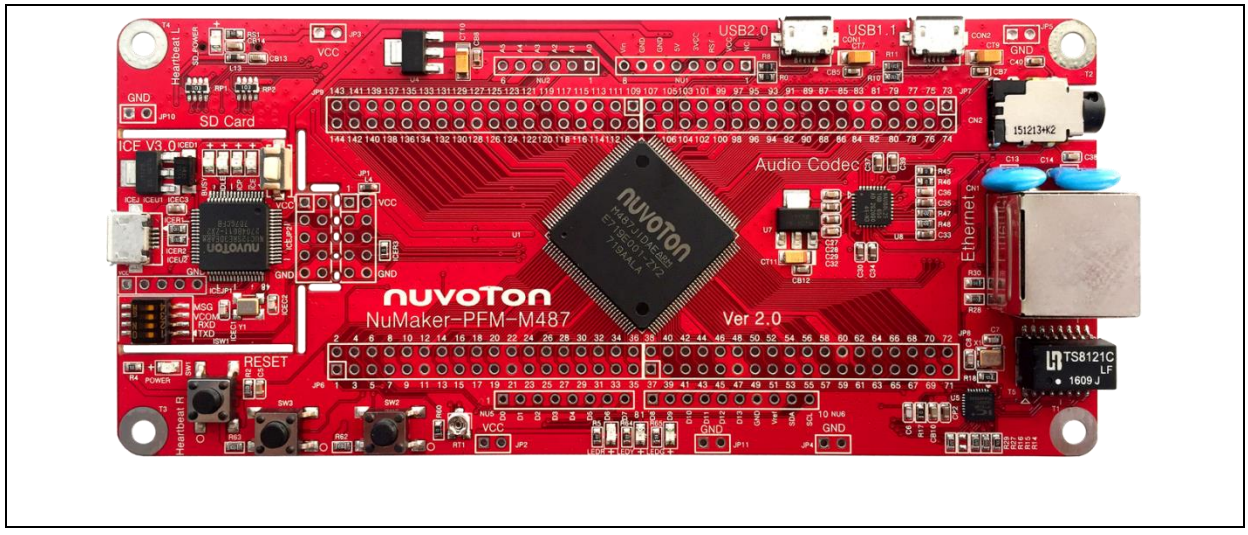


Figure 1-1 NuMaker-PFM-M487 Board

The left portion of this board is the M487 Platform that includes the target chip M487 MCU which embedded ARM® Cortex®-M4 core with DSP extensions and a Floating Point Unit (FPU) and the other related on-board application parts and connectors.

The right portion of this board is a Nu-Link-Me ICE Bridge based on the SWD (Serial Wire Debug) interface connected with the target chip, allowing user to program the application code to the flash of target chip through the USB port from PC Host.

For more information on the NuMaker-PFM-M487 board for the ARM® mbed™ Device Platform, please visit the ARM® mbed™ hardware board website:

https://developer.mbed.org/platforms/NUMAKER-PFM-M487/

1.3 Introduction to ARM® mbed™

The ARM® mbed™ IoT Device Platform provides the operating system, cloud services, tools and developer ecosystem to make the creation and deployment of commercial, standards-based IoT solutions possible.

The ARM® mbed™ allows IoT devices to collaborate and communicate with each other on the basis of transparency - otherwise each of the devices will not be able to talk to each other or to the cloud. For more detailed information about ARM® mbed™, user can visit the related ARM® mbed™ websites as the follows:

ARM® mbed™ homepage: www.mbed.com/en/

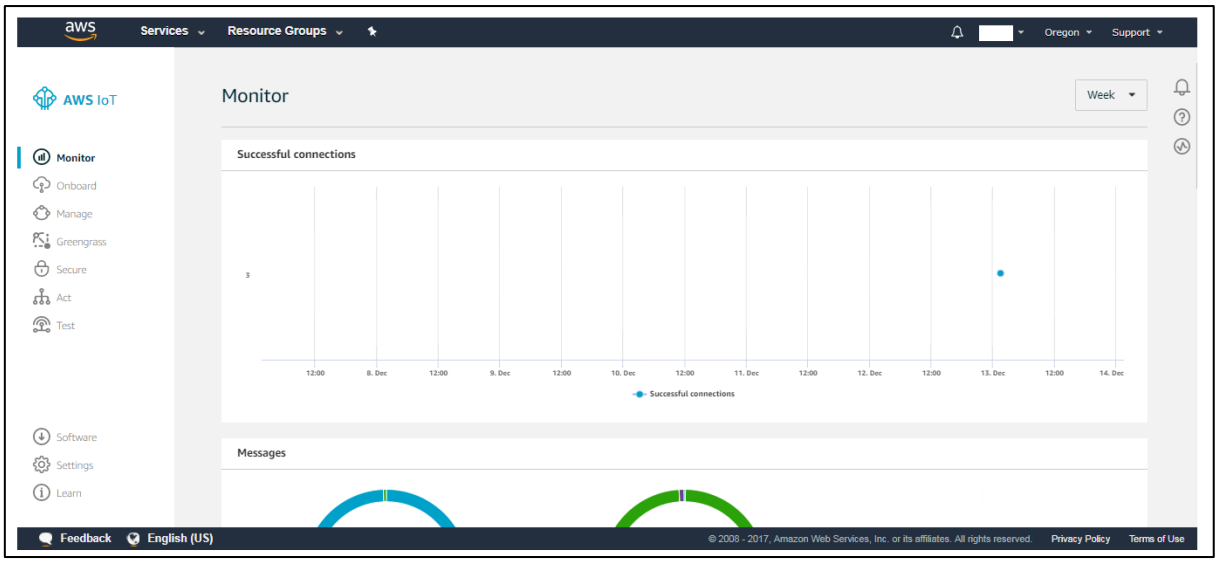
Software homepage: docs.mbed.com/docs/mbed-os-handbook/en/

Official C/C++ SDK: developer.mbed.org/users/mbed_official/code/mbed/

2 CONFIGURING THE AWS IOT SERVICE

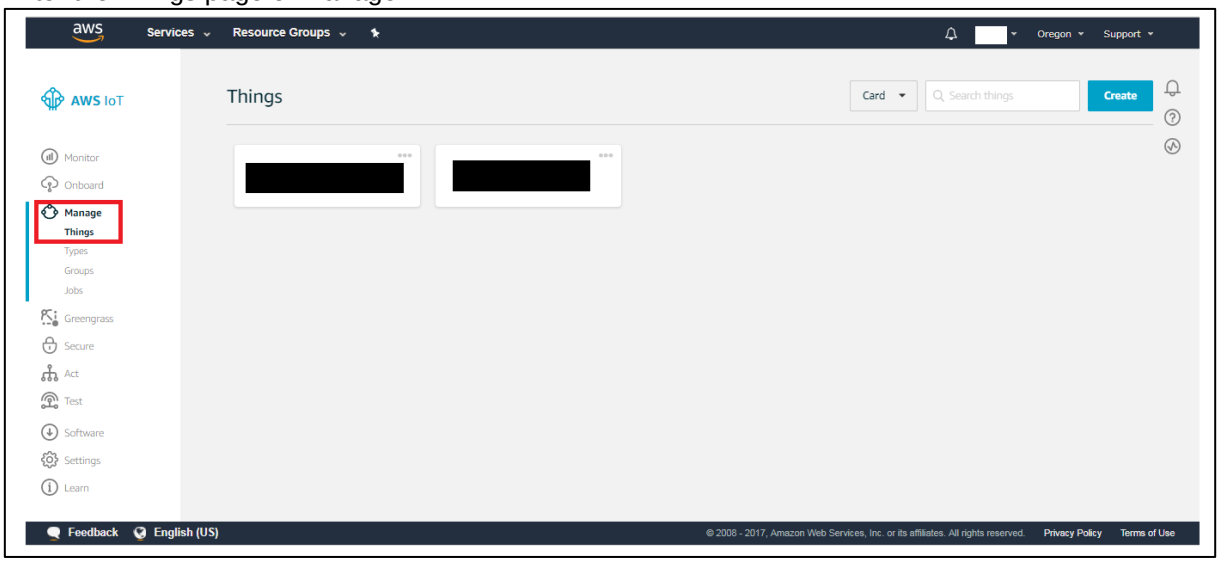
2.1 Create AWS IoT Account

- Visit AWS IoT website: https://aws.amazon.com/ to create a account for AWS IoT
- Sign in the console and select the service of AWS IoT to enter the homepage of AWS IoT service

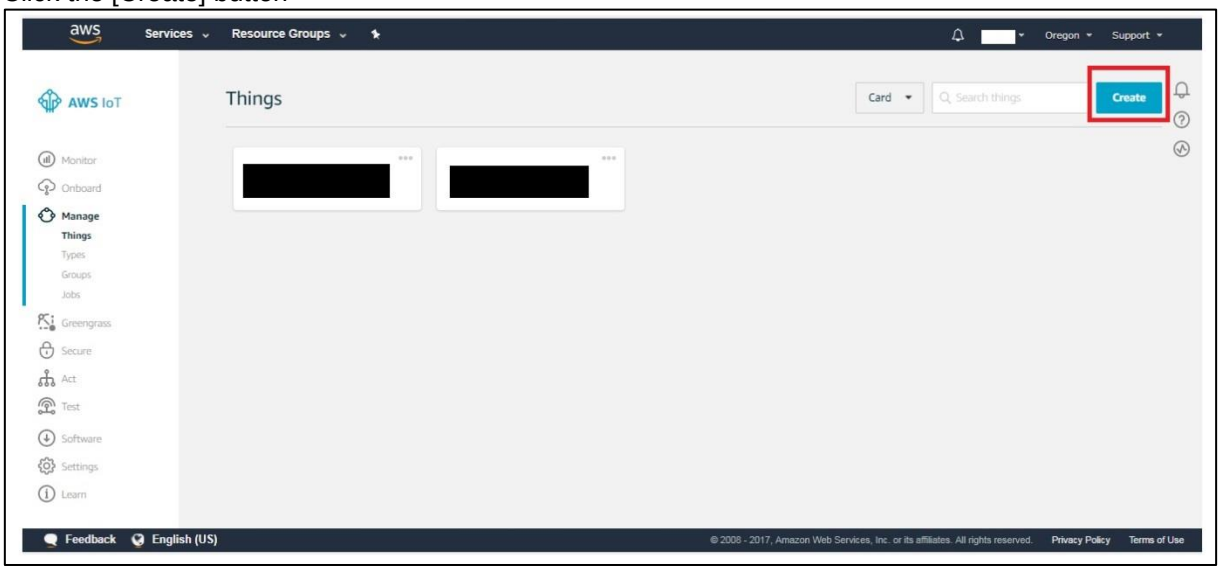


2.2 Create Thing

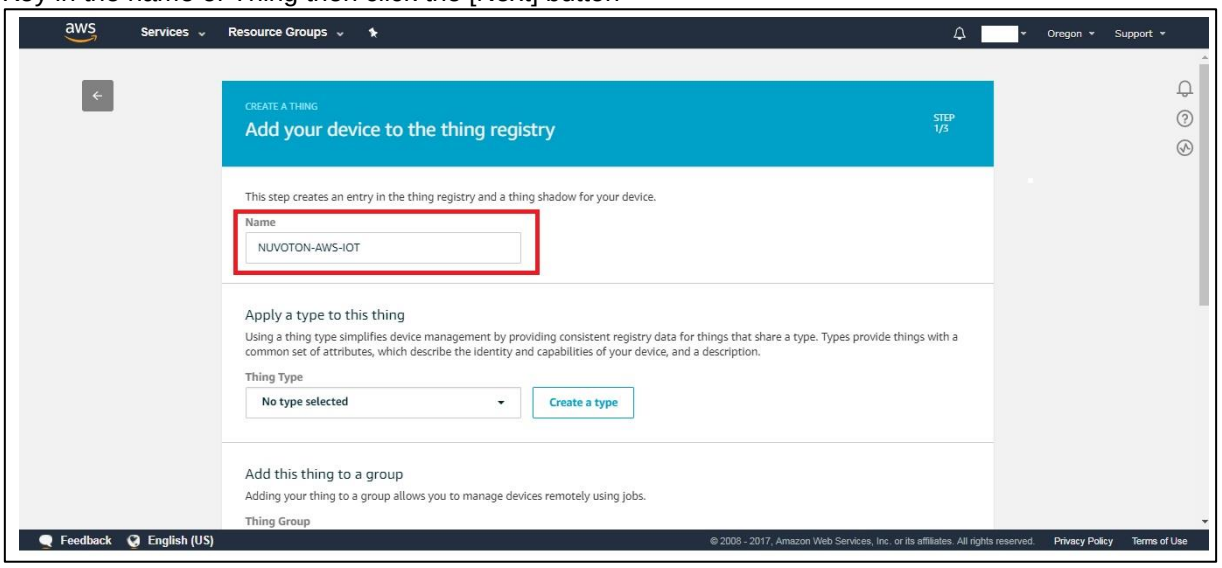
- Enter the Things page of manage



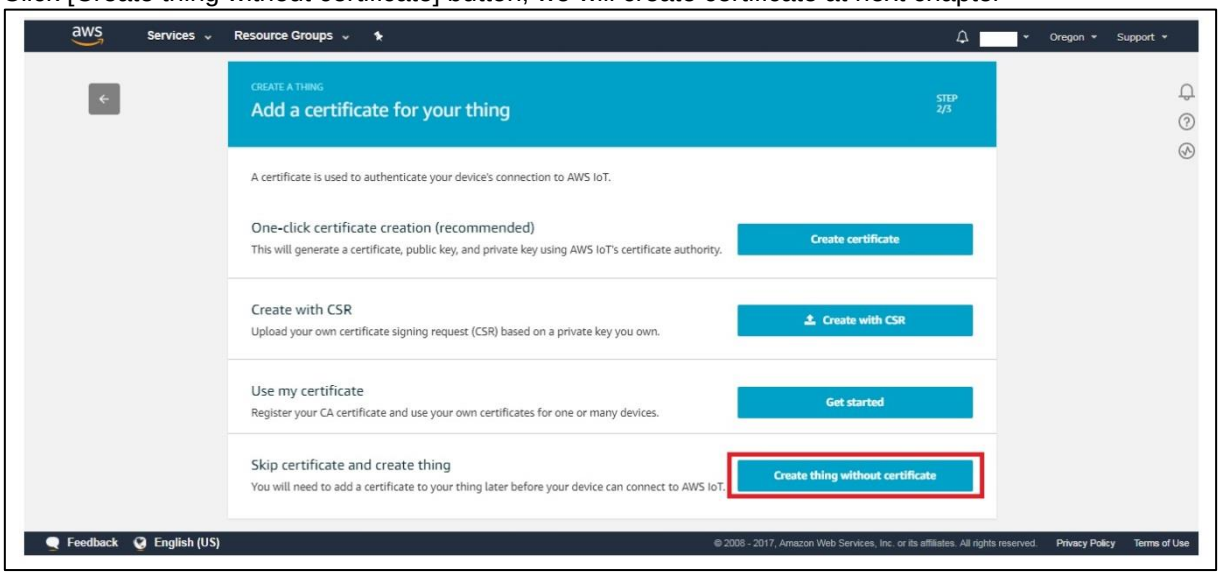
✧ Click the [Create] button



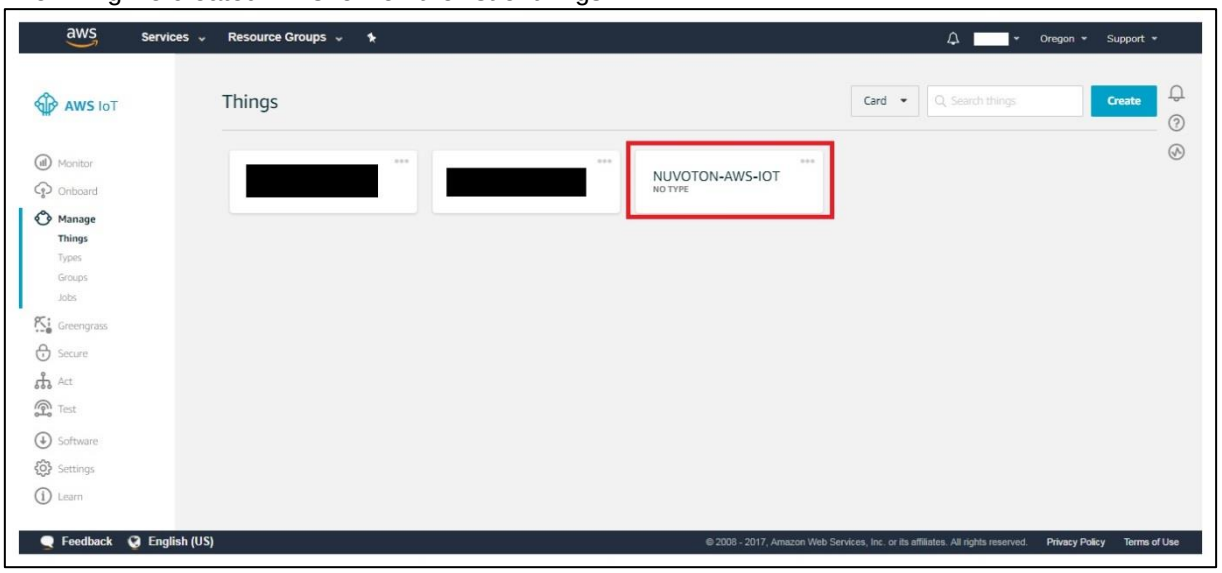
✧ Key in the name of Thing then click the [Next] button



✧ Click [Create thing without certificate] button, we will create certificate at next chapter

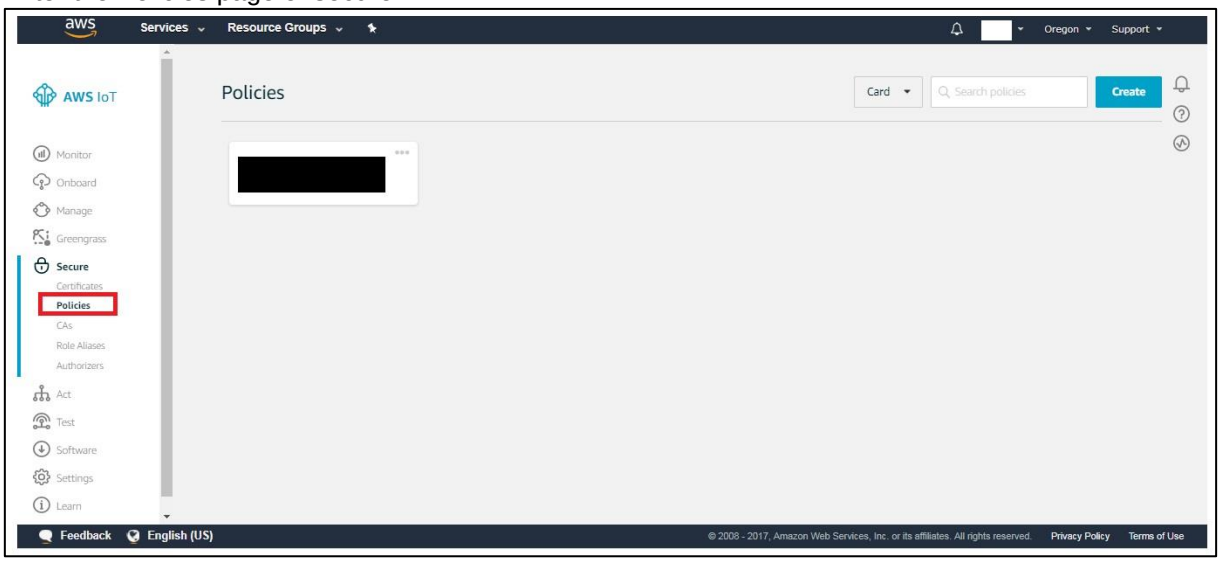


✧ The Thing we created will show on the list of things

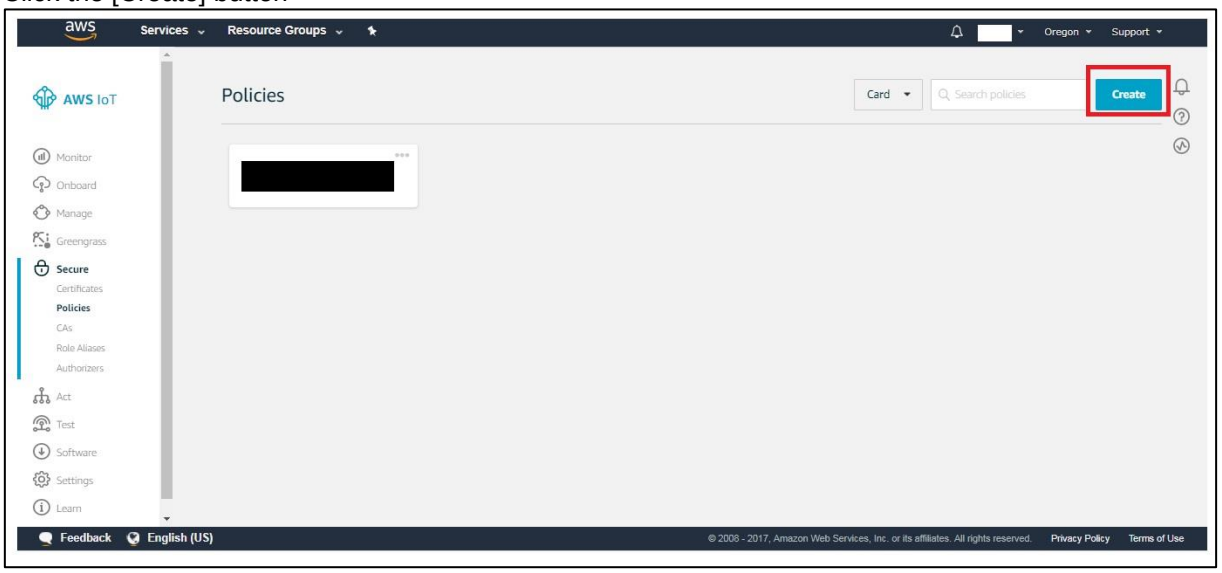


2.3 Create Policies

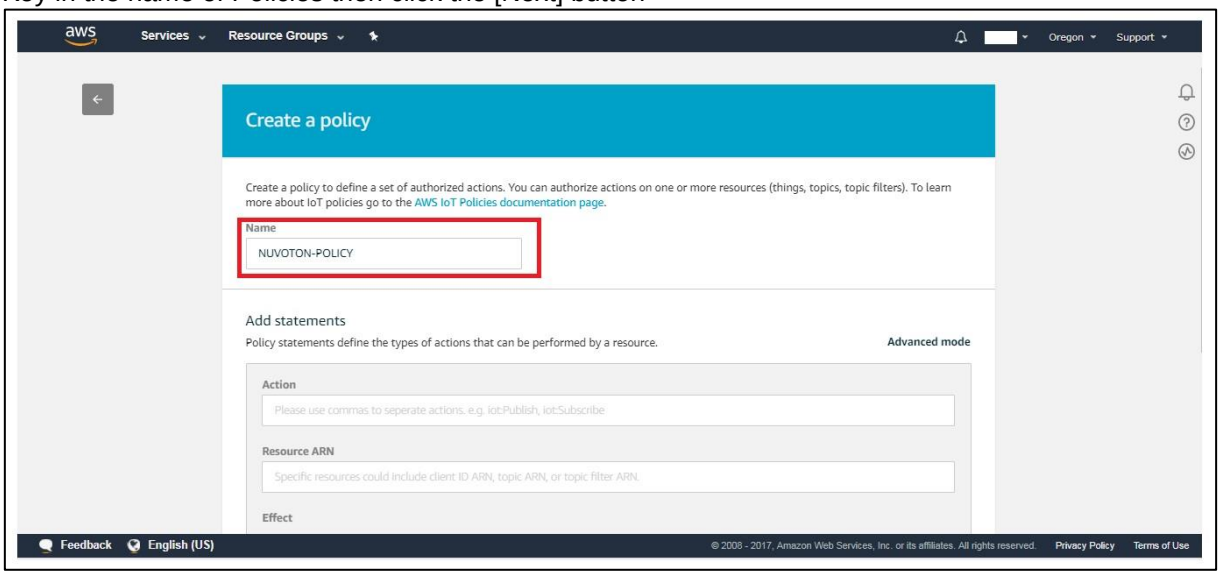
✧ Enter the Policies page of secure



✧ Click the [Create] button



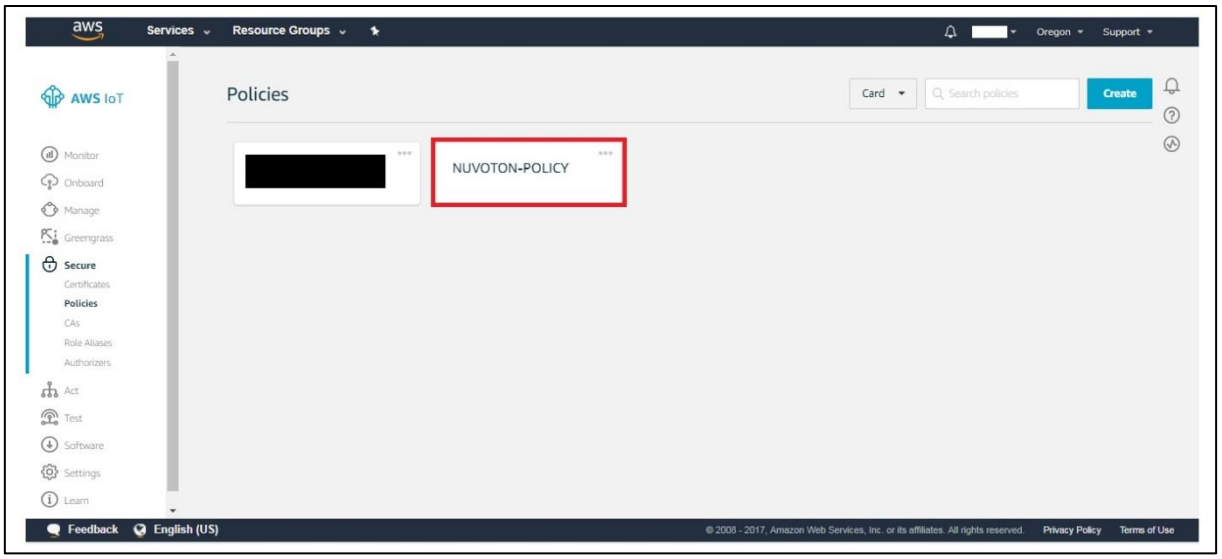
✧ Key in the name of Policies then click the [Next] button



✧ Change to [Advanced mode] and modify statements then click [Create] button

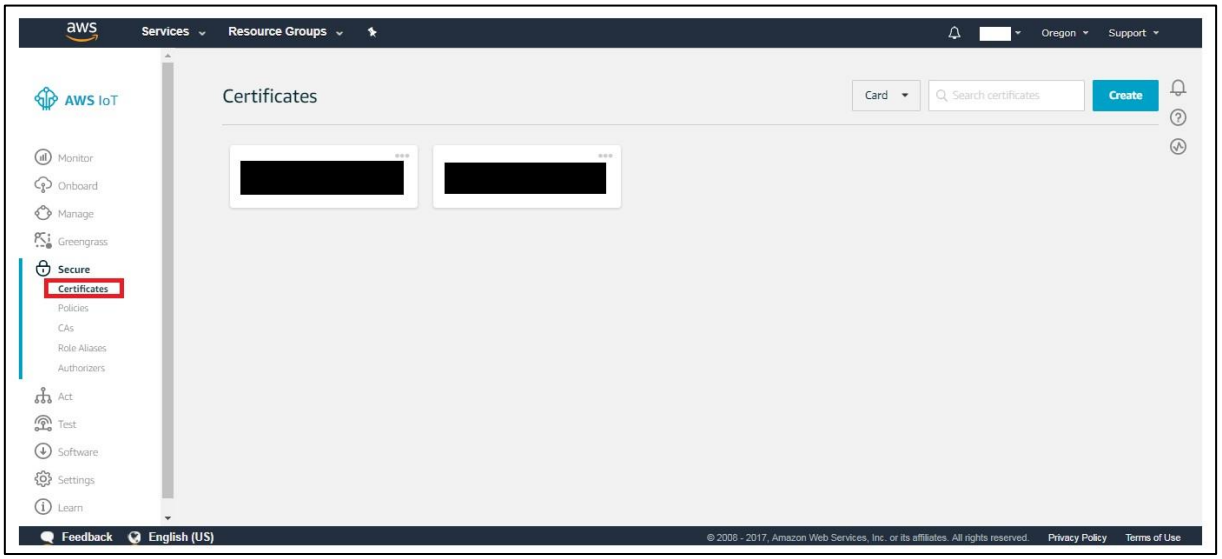
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*"
    }
  ]
}
```

✧ The Policy we created will show on the list of policies

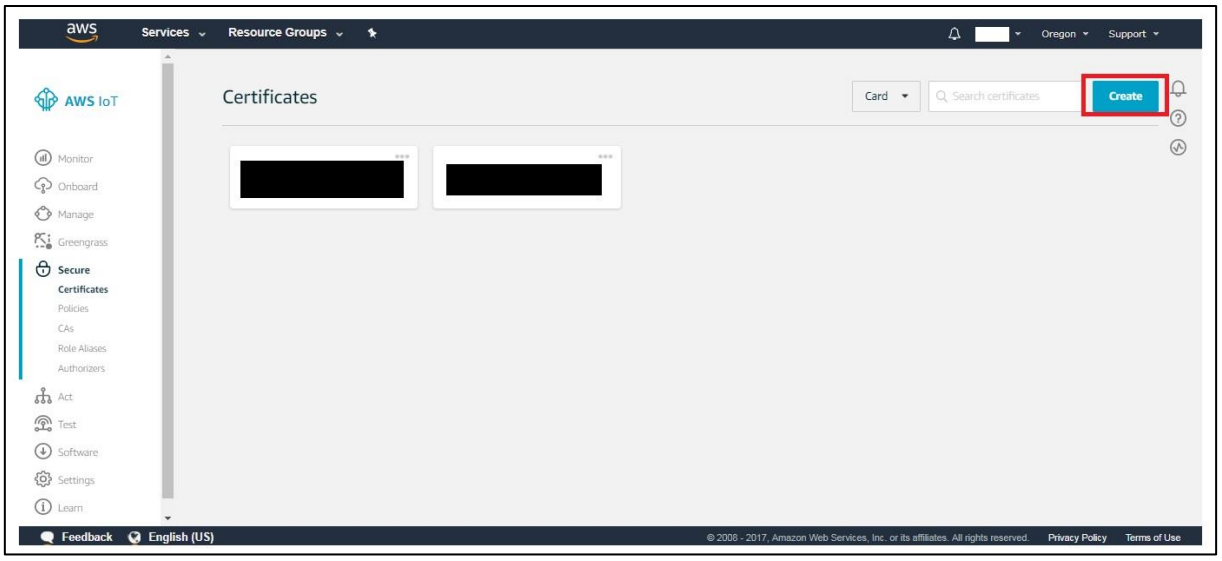


2.4 Create Certificate

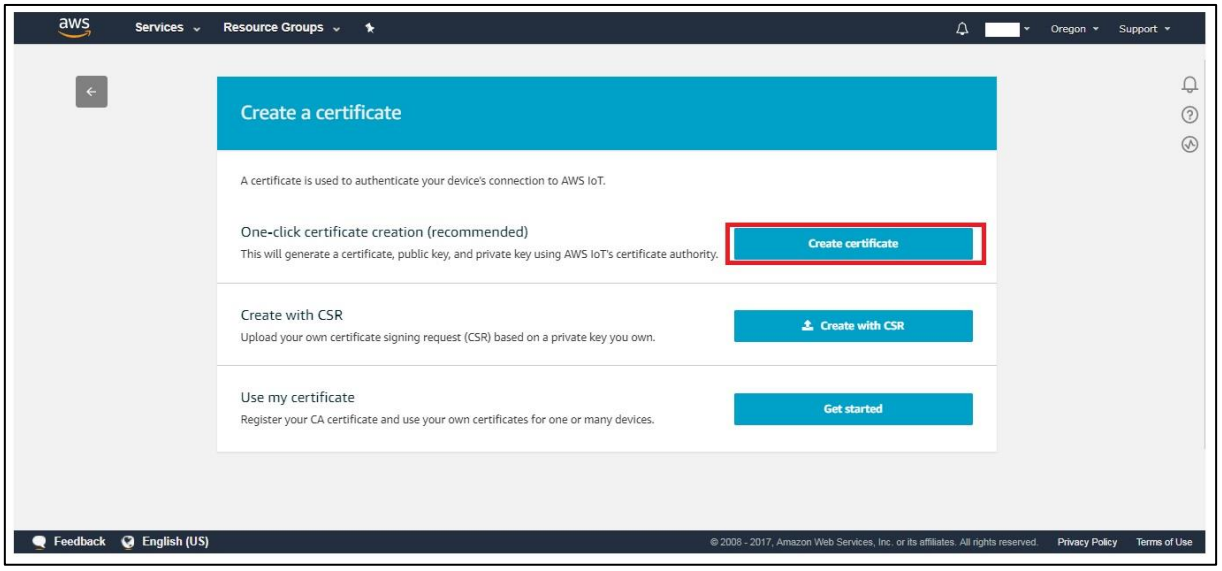
✧ Enter the Certificate page of secure



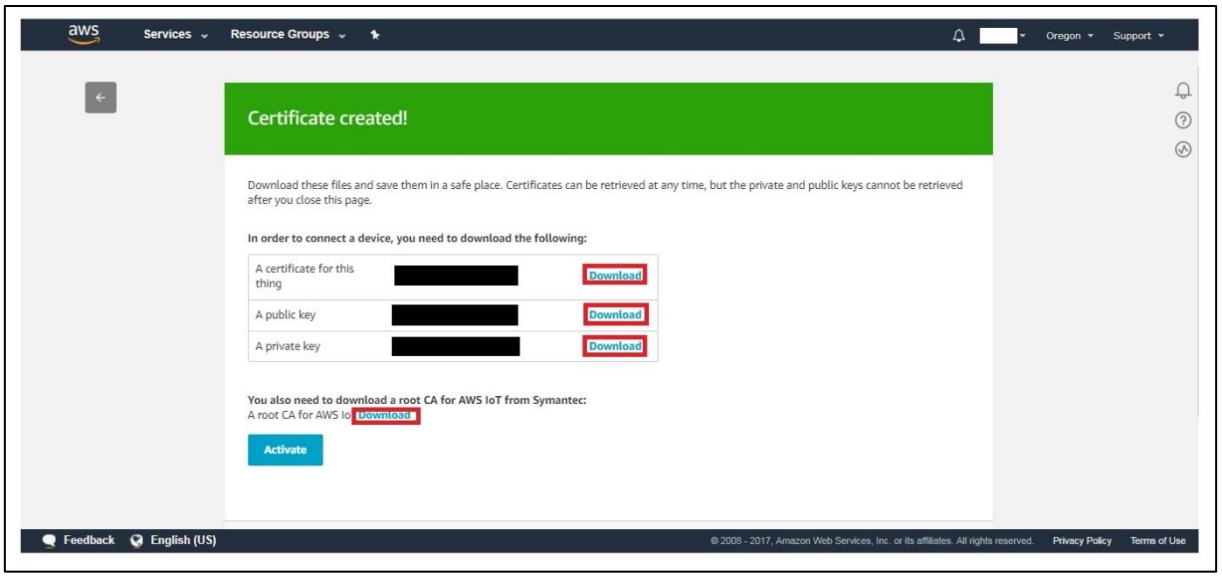
✧ Click the [Create] button



✧ Click the [Create certificate] button

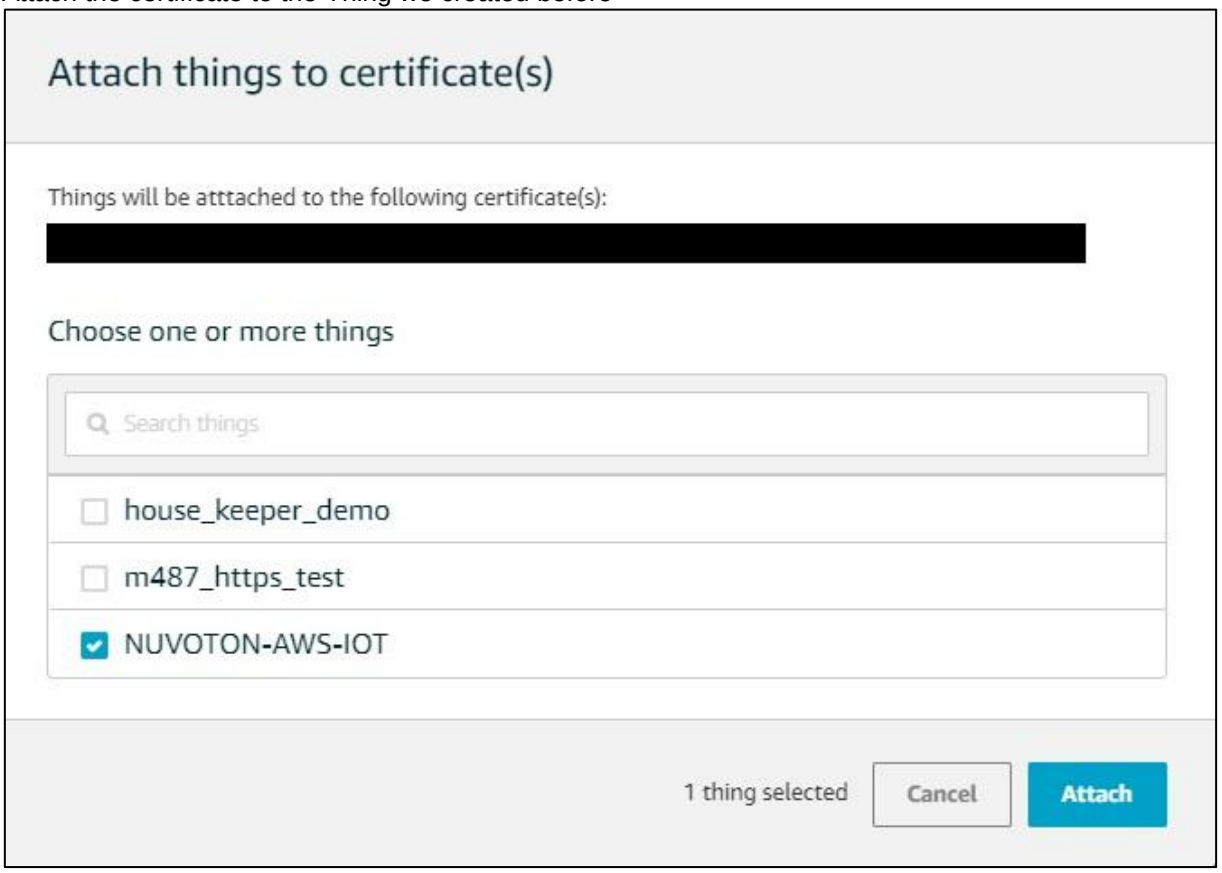


✧ Download the certificate, public key, private key and root CA then click the [Active] button



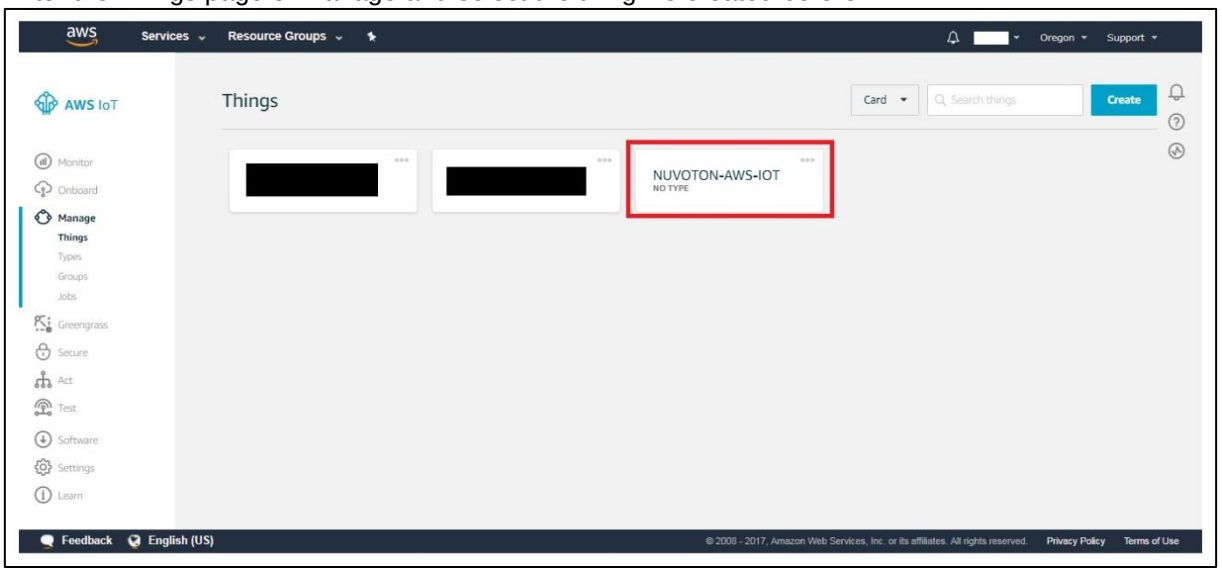
✧ Attach the certificate to the policy we created before

✧ Attach the certificate to the Thing we created before



2.5 Things Shadow for AWS IoT

✧ Enter the Things page of manage and select the thing we created before



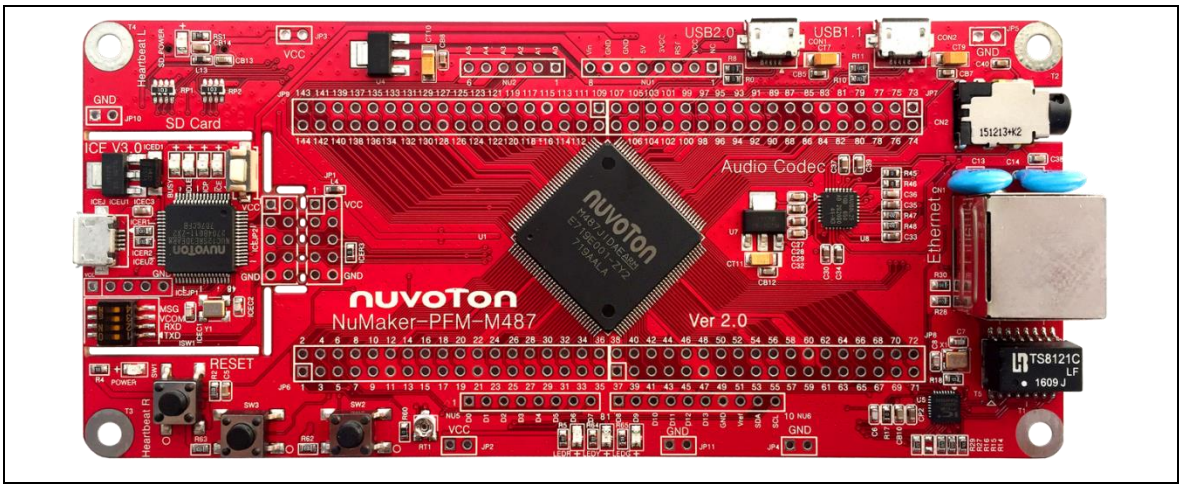
✧ Enter the shadow page and modify the shadow document

```
{
    "state": {
        "desired": {
            "LED1": "0",
            "LED2": "0",
            "LED3": "0"
        },
        "reported": {
            "LED1": "0",
            "LED2": "0",
            "LED3": "0"
        }
    }
}
```

3 SETUP NUMAKER-PFM-M487

3.1 Hardware Requirement

✧ The NuMaker-PFM-M487

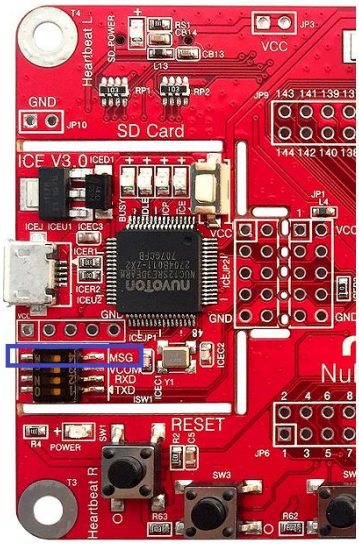


✧ The ESP8266 Wi-Fi Module x1

✧ The Wi-Fi AP station to share network

3.2 Hardware Setting

✧ Enable [Mass Storage Update] function by dip switch



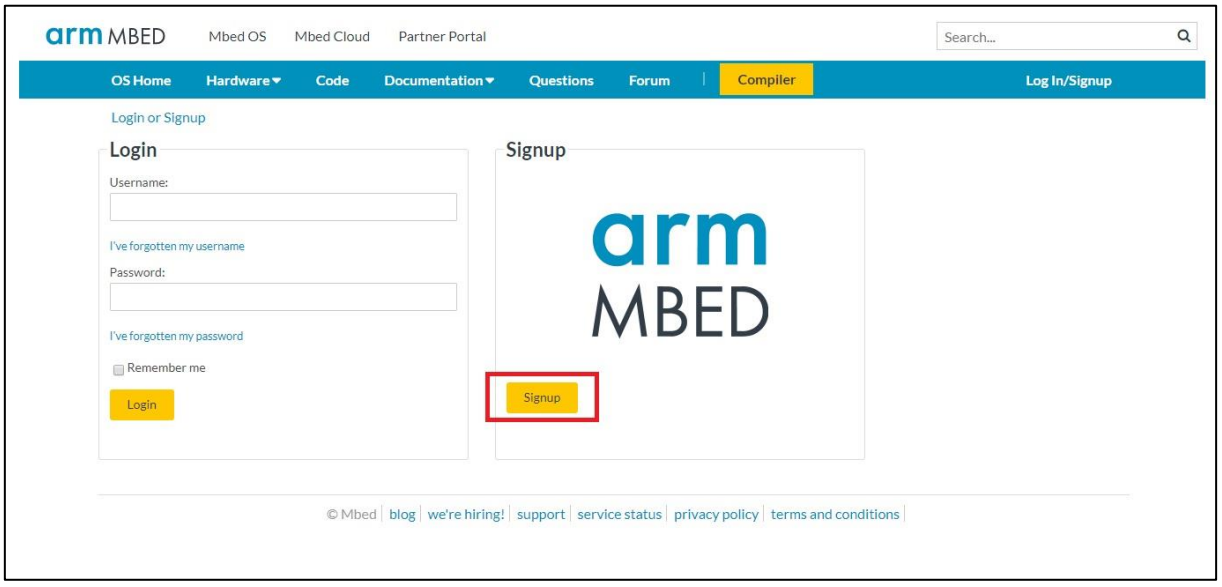
✧ Connect ESP8266 Wi-Fi module to NuMaker-PFM-M487

NuMaker-PFM-M487	ESP8266
VDD	VDD
GND	GND
UART_TX	RX
UART_RX	TX

4 NUVOTON-AWS-IOT

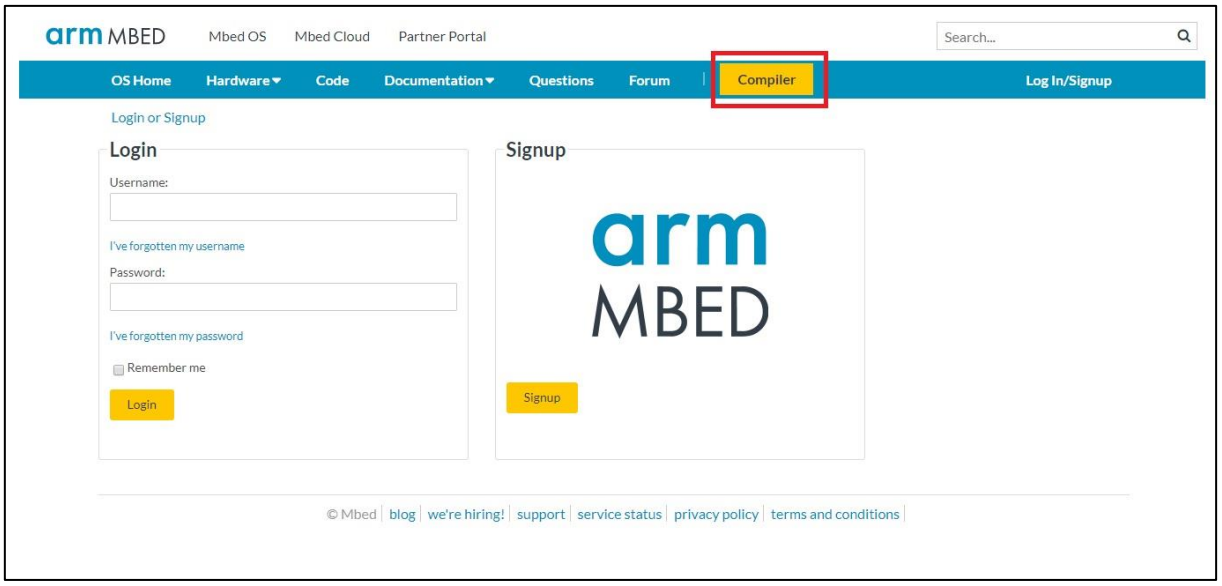
4.1 Create ARM® mbed™ Account

✧ Visit ARM® mbed™ website: https://os.mbed.com/ to create a account for ARM® mbed™



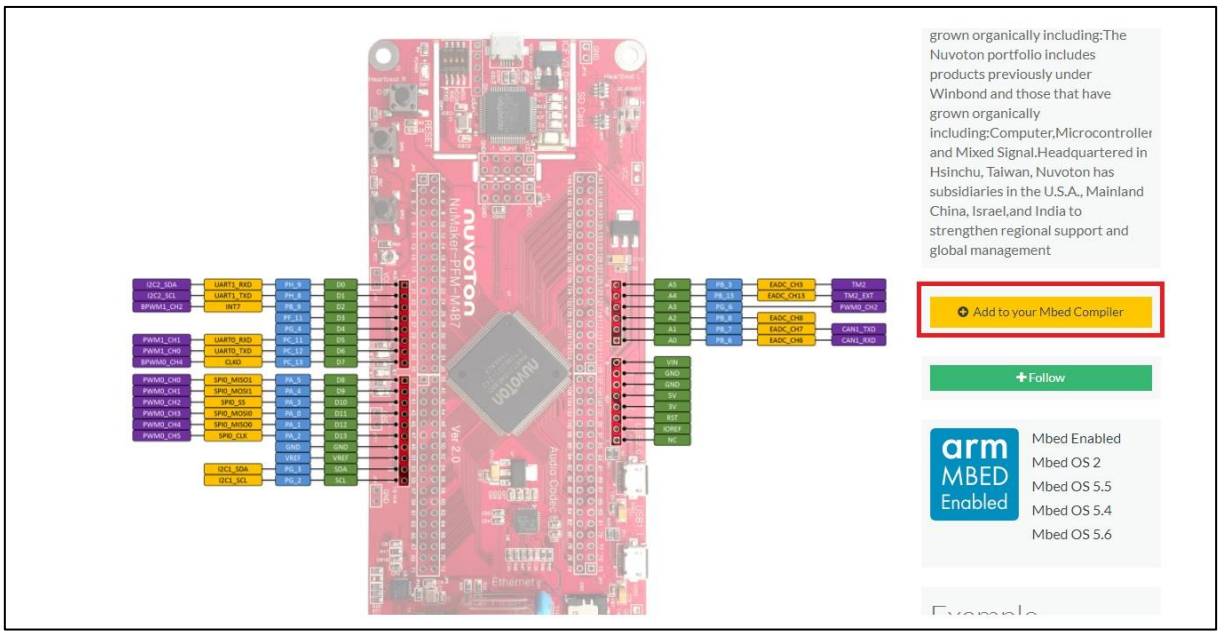
4.2 Import [Nuvoton-AWS-IoT] to compile

✧ Sign in the on-line compile



✧ Add NuMaker-PFM-M487 platform to on-line compile

Visit https://os.mbed.com/platforms/NUMAKER-PFM-M487/ and import NuMaker-PFM-M487 platform to on-line compiler



✧ Import NUVOTON-AWS-IOT project

補圖

4.3 Modify the Setting of [mbed_app.json]

✧ Open the [mbed_app.json] file

✧ Change the Wi-Fi SSID and password according network setting

```
"wifi-ssid": {
    "value": "\"NUVOTON\""
},
"wifi-password": {
    "value": "\"123456789\""
}
```

✧ Change the ESP8266 pin according hardware setting

```
"esp8266-tx": {
    "help": "Pin used as TX (connects to ESP8266 RX)",
    "value": "D1"
},
"esp8266-rx": {
    "help": "Pin used as RX (connects to ESP8266 TX)",
    "value": "D0"
},
```

4.4 Modify the Configurations of [NUVOTON-AWS-IOT]

✧ Open the [main.c] file

✧ Setting the root CA based on the root CA file downloaded at chapter 2.4

```
const char SSL_CA_CERT_PEM [] = \
"-----BEGIN CERTIFICATE-----\n"\
"MIIE0zCCA7ugAwIBAgIQGNr[redacted]\n"\
"yjELMAkGA1UEBhMCVVMxFzA[redacted]\n"\
.
.
.
.
"biwgSW5jLiAtIEZvciBhdXR[redacted]\n"\
"U2lnbiBDbGFzcyAzIFB1Ymx[redacted]\n"\
"-----END CERTIFICATE-----\0";
```

✧ Setting the certificate based on the certificate file downloaded at chapter 2.4

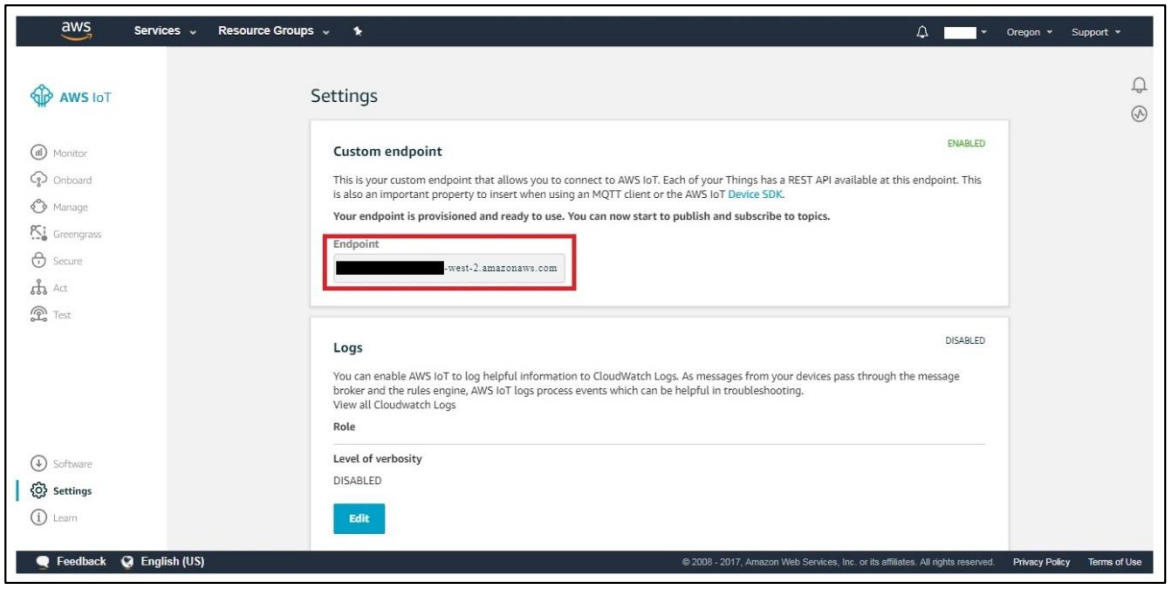
```
const char SSL_USER_CERT_PEM [] = \
"-----BEGIN CERTIFICATE-----\n"\
"MIIDWjCCAkKgAwIBAgIVAI[redacted]\n"\
"CwUAME0xSzBJBgNVBAsMQk[redacted]\n"\
.
.
.
.
"Hdcoxz+ymWr5CFsmsquey+[redacted]\n"\
"w31E6OhnrAMXnGFwipLFVG[redacted]\n"\
"-----END CERTIFICATE-----\0";
```

✧ Setting the private key based on the private key file downloaded at chapter 2.4

```
const char SSL_USER_PRIV_KEY_PEM [] = \
"-----BEGIN RSA PRIVATE KEY-----\n"\
"MIIEpAIBAAKCAQEApbEEGN[redacted]\n"\
"4DEo6NHuclgVxFLCzjQb5E[redacted]\n"\
.
.
.
.
"WKbtCHvmP92LHJv3sG/g9h[redacted]\n"\
"MG9fXtNCZz1xajUMRBSxQLs[redacted]\n"\
"-----END RSA PRIVATE KEY-----\0";
```

✧ Setting the URI of RESTful API

```
#define AWS_IOT_HTTPS_SERVER_NAME "[redacted].iot.us-west-2.amazonaws.com"
```

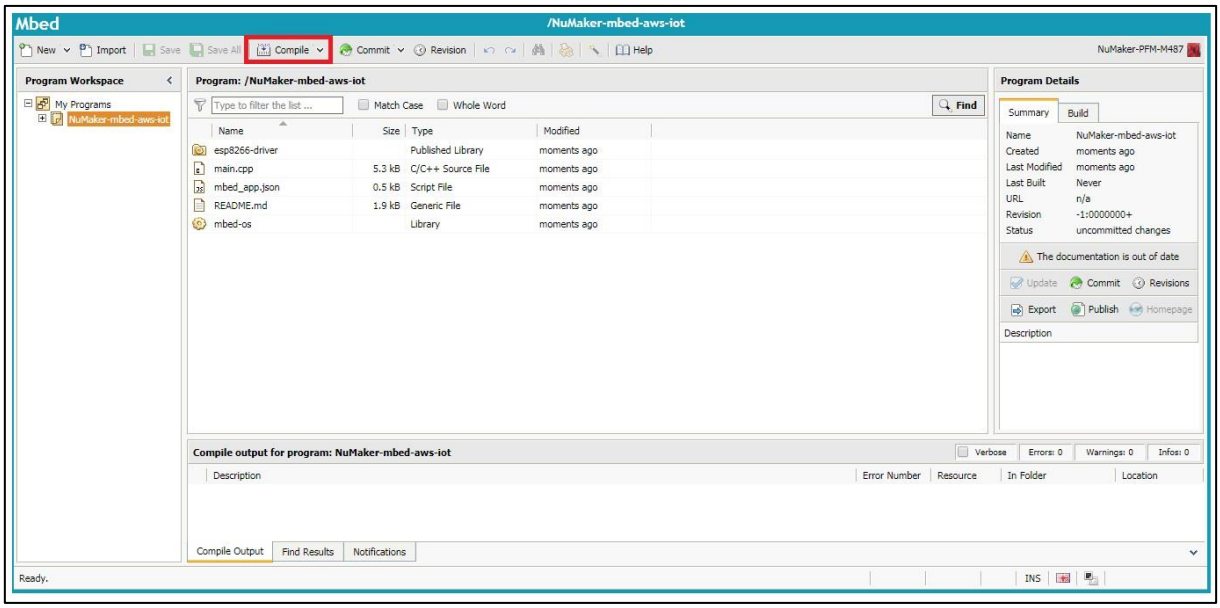


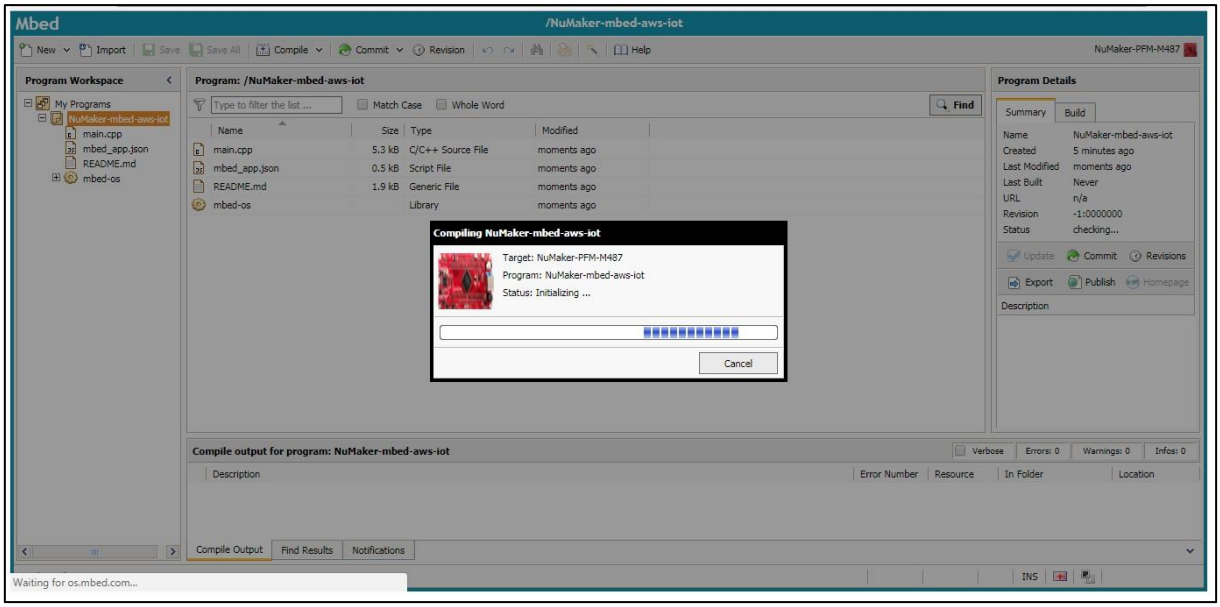
✧ Setting the name of thing to [NUVOTON-AWS-IOT]

```
#define AWS_IOT_HTTPS_THINGNAME      "NUVOTON-AWS-IOT"
```

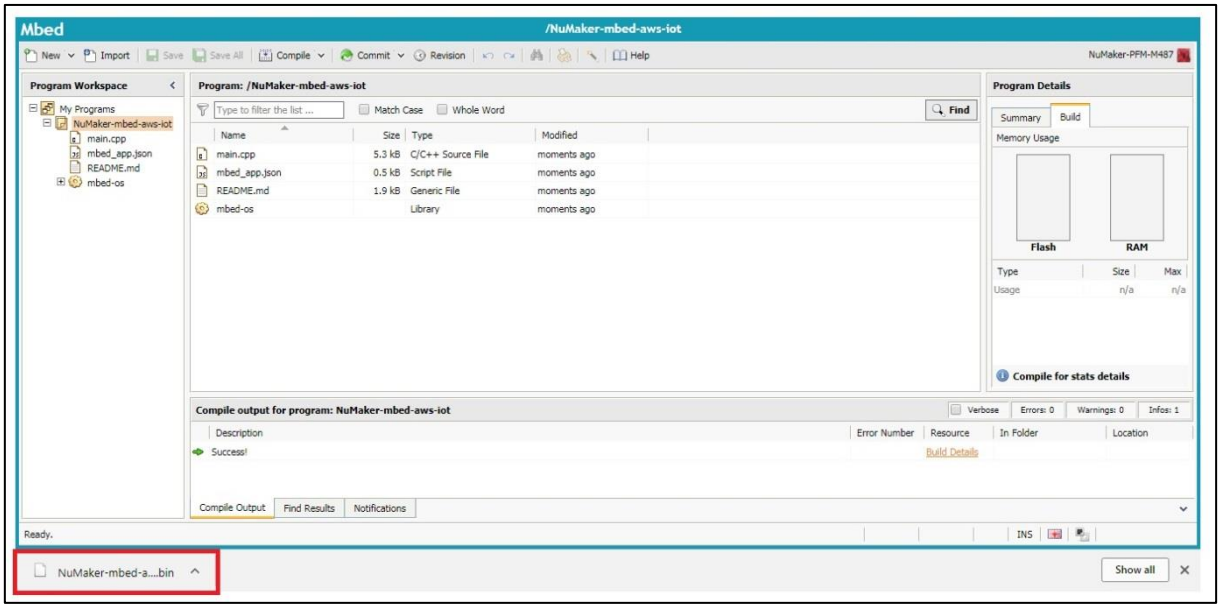
4.5 Compile and Install

✧ Click the [Compile] on the on-line compiler



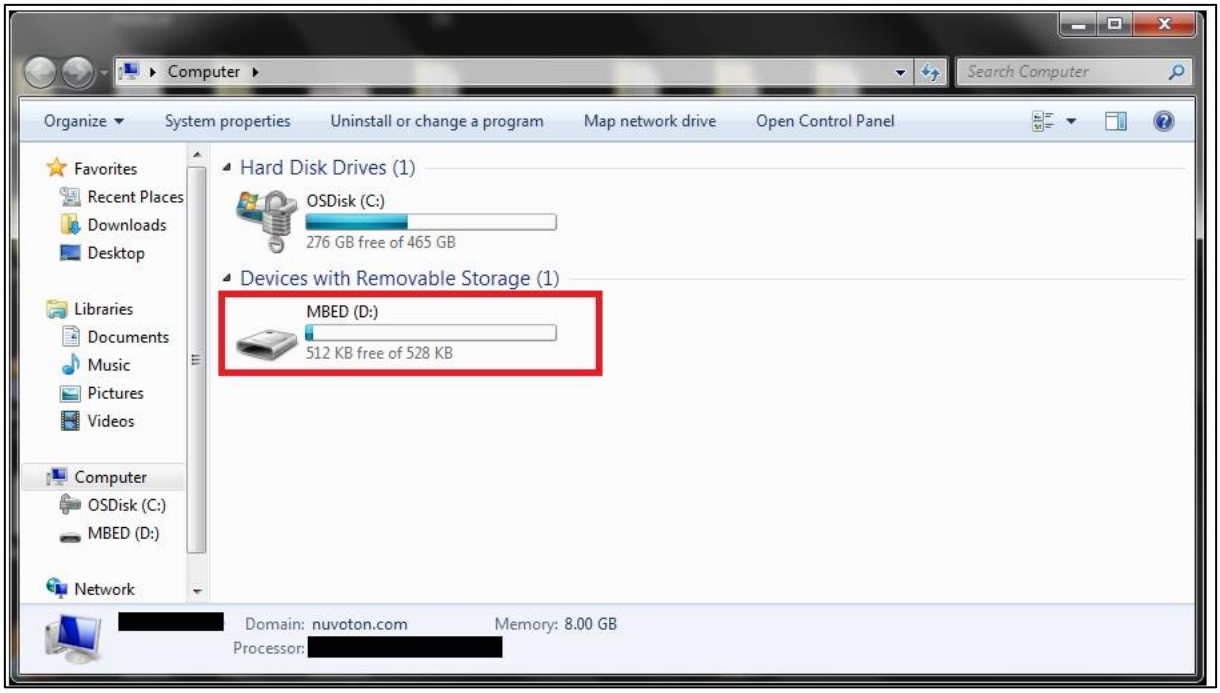


When the compiled file is ready, it's downloaded to your default download location.



✧ Install

Connect NuMaker-PFM-M487 to computer over USB. NuMaker-PFM-M487 is shown as “MBED” removable storage.



Drag and drop your program to the NuMaker-PFM-M487. The NuMaker-PFM-M487 installs the program.

5 TEST NUVOTON-AWS-IOT

✧ Enter the shadow page of [NUVOTON-AWS-IOT] on the AWS IoT console

✧ Change the led status in the desired of shadow state, the NuMaker-PFM-M487 will follow the shadow state to on/off the LEDs.

6 REVISION HISTORY

Date	Revision	Description
2019.04.15	1.00	1. Initially issued.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.