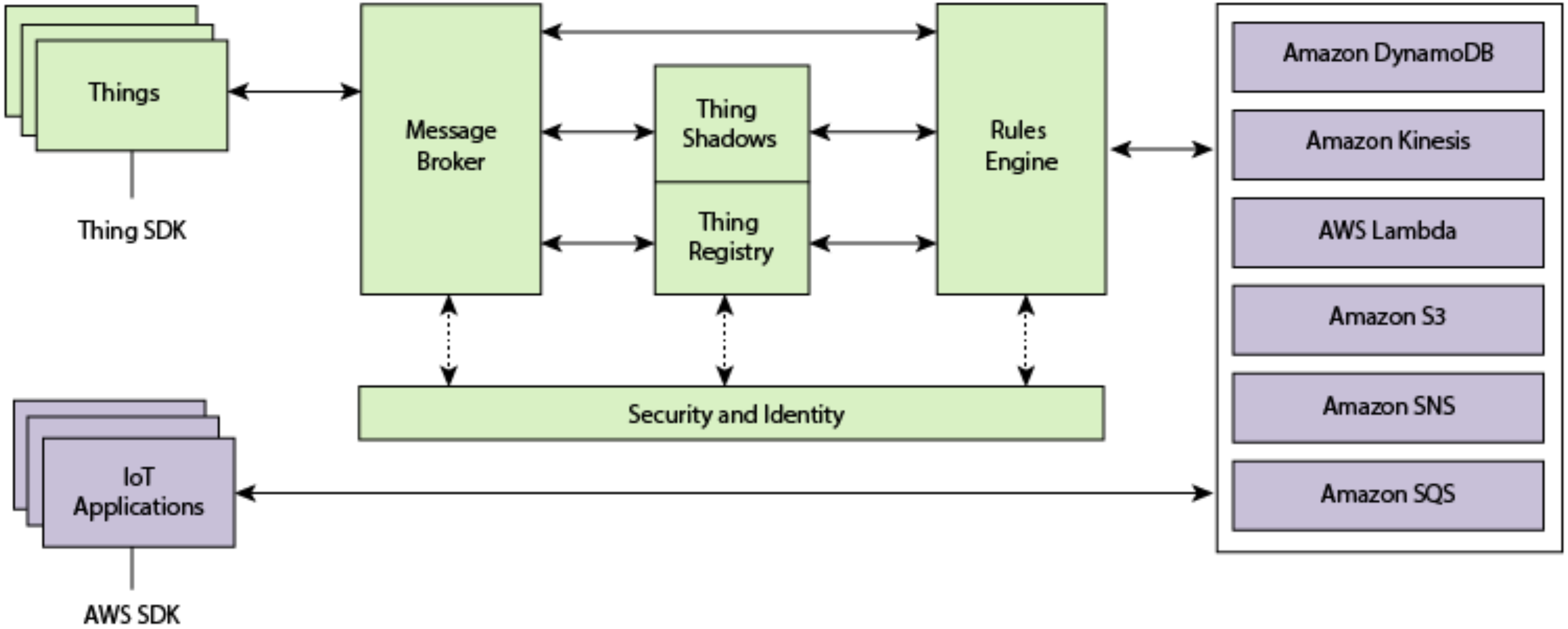
AWS IoT Workshop with LinkIt One

John Chang
Ecosystem Solution Architect
johnchan@amazon.com

What Is AWS IoT

- AWS IoT is considered a beta service as defined in the Service Terms



MediaTek LinkIt™ ONE and Grove IoT Starter Kit Powered by AWS

- Product Page
  - https://labs.mediatek.com/site/global/developer_tools/mediatek_linkit/hdk_intro/aws_kit/index.gsp
- LinkIt ONE Connect to AWS IoT Services
  - https://github.com/MediaTek-Labs/aws_mbedtls_mqtt

MediaTek LinkIt™ ONE and Grove
IoT Starter Kit Powered by AWS
If you can write
"HELLO, WORLD!"
you can change the world.
Most of what you need to make a difference
in IoT development is already inside your head.
The rest is inside this box.
LinkIt
amazon
seeed

Get Started with AWS IoT Services on the LinkIt™ ONE Development Platform

- Setup for development on LinkIt ONE using the Arduino IDE as described here.
- Install and configure the AWS IoT services command line interface (CLI) and a MQTT client.
- Add the mbed TSL library for LinkIt ONE to your Arduino IDE.
- Provision your AWS certificates to the LinkIt ONE development board.
- Create your LinkIt ONE application and run it on your board.

Setup for development on LinkIt ONE using the Arduino IDE

Get Started with the LinkIt™ ONE Development Platform

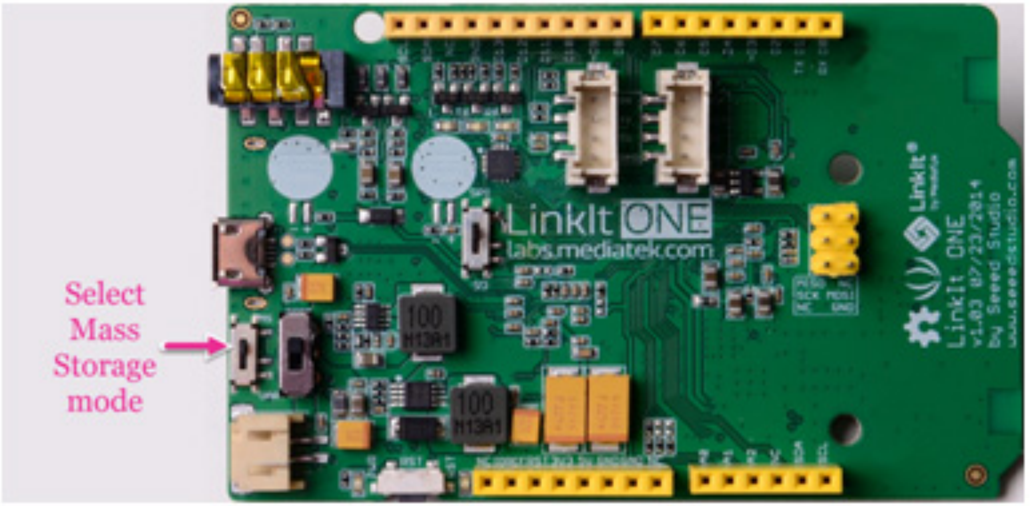
- Install Arduino 1.5.7
  - https://www.arduino.cc/en/Main/OldSoftwareReleases#1.5.x
- Windows PC
  - **Download the Arduino IDE**
  - **Download the LinkIt ONE SDK**
- Mac
  - **Download the Arduino IDE**
  - **Download the LinkIt ONE SDK** (LinkIt ONE SDK can be used on Mac OSX 10.09 and 10.10)

Install the Arduino IDE and LinkIt ONE SDK

- Windows PC
  - https://labs.mediatek.com/site/global/developer_tools/mediatek_linkit/get-started/windows/install/
- MAC
  - https://labs.mediatek.com/site/global/developer_tools/mediatek_linkit/get-started/mac/install/

Update Your Board’s Firmware

- Set the board to Mass Storage Bootup mode



- Run the LinkIt ONE Firmware Updater Application
  - https://labs.mediatek.com/site/global/developer_tools/mediatek_linkit/get-started/windows/update/

Configure the Arduino IDE

- Windows PC
  - https://labs.mediatek.com/site/global/developer_tools/mediatek_linkit/get-started/windows/configure/
- MAC
  - https://labs.mediatek.com/site/global/developer_tools/mediatek_linkit/get-started/mac/configure/
- Create and Run Your First Sketch
  - n Arduino IDE on the **File** menu point to **Examples**, then **01. Basics** and click **Blink**, this opens the Blink example.

Install and configure the AWS IoT services command line interface (CLI) and a MQTT client

Create a Device in the Thing Registry

- Install the AWS CLI
  - Getting Set Up with the AWS Command Line Interface.
- Create a thing named ”XXXXX"
  - **aws iot create-thing --thing-name** *“XXXXX”*
- Confirm the thing has been created
  - **aws iot list-things**

Secure Communication Between a Device and AWS IoT

- Provision a Certificate
  - **aws iot create-keys-and-certificate --set-as-active --output text**
  - output of the command contains the certificate, the certificate's ARN, the public key, and the private key.
- Save the certificate to a file
  - **aws iot describe-certificate --certificate-id "*certificate-id*" --output text --query certificateDescription.certificatePem > cert.pem**
  - The certificate-id is part of the certificate ARN

Secure Communication Between a Device and AWS IoT

- Create and Attach an AWS IoT Policy to Your Certificate
  - Policy Document

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

- Create the AWS IoT policy
  - **aws iot create-policy --policy-name "***AllowAllIotOperationsPolicy***" --policy-document** file://*AllowAllIotOperations.json*

Secure Communication Between a Device and AWS IoT

- Attach the policy to your certificate
  - **aws iot attach-principal-policy --principal "***certificate-arn***" --policy-name "***AllowAllIotOperationsPolicy***"**
- Attach your Certificate to Your Device
  - **aws iot attach-thing-principal --thing-name** *”XXXXX"* **--principal** "*certificate-arn*"

Setting Up CloudWatch Logs

- Create an IAM Role for Logging
  - Role policy

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "logs:CreateLogGroup",
            "logs:CreateLogStream",
            "logs:PutLogEvents",
            "logs:PutMetricFilter",
            "logs:PutRetentionPolicy"
         ],
         "Resource": [
            "*"
         ]
      }
   ]
}
```

Setting Up CloudWatch Logs

- Create an IAM Role for Logging
  - Trust policy

```
{
 "Version": "2012-10-17",
  "Statement": [
   {
     "Sid": "",
     "Effect": "Allow",
     "Principal": {
       "Service": "iot.amazonaws.com"
     },
     "Action": "sts:AssumeRole"
   }
  ]
}
```

Setting Up CloudWatch Logs

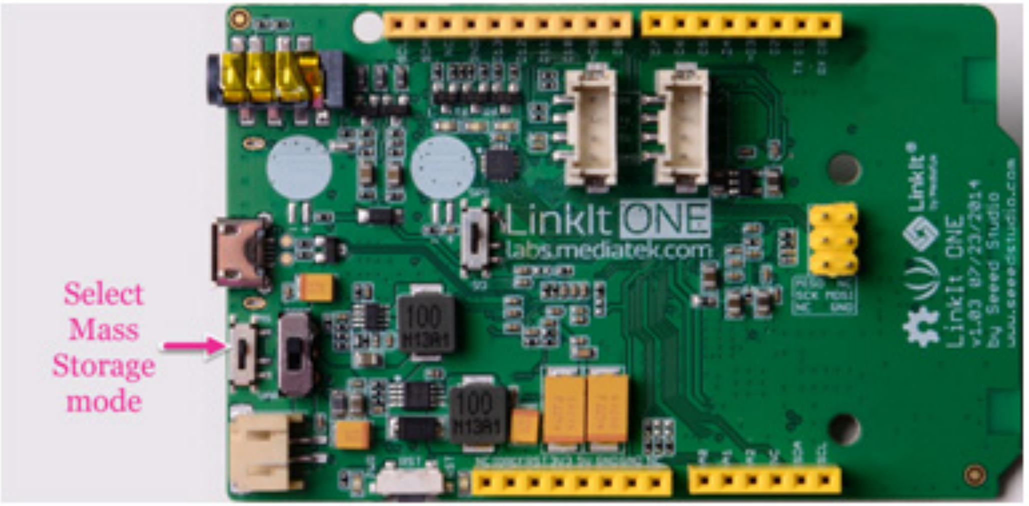
- Register the Logging Role with AWS IoT
  - **aws iot set-logging-options --logging-options-payload roleArn=**”arn:aws:iam::*<your-aws-account-num>*:role/IoTLoggingRole”,**logLevel=**”INFO”
- Log level
  - **DEBUG** provides the most detailed information of AWS IoT activity.
  - **INFO** provides a summarized view of most actions. This is sufficient for most users.
  - **ERROR** provides error cases only.
  - **DISABLED** removes logging altogether, but keeps your logging role intact.

Add the mbed TSL library for LinkIt ONE to your Arduino IDE

Create and run your first AWS IoT sketch

Provision your AWS certificates to the LinkIt ONE development board

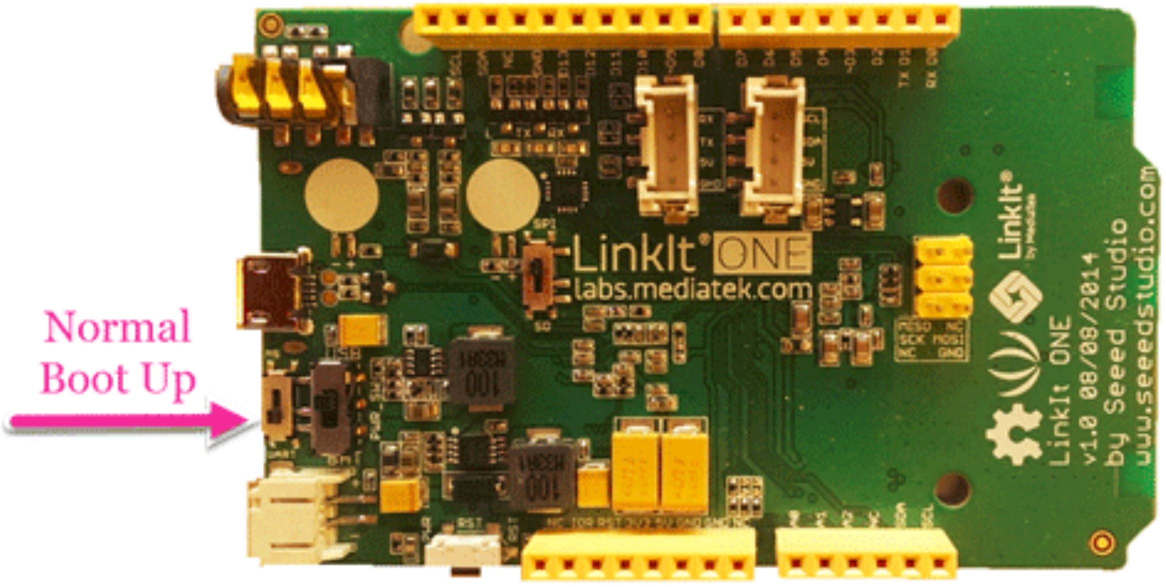
- Copy the .pem files into the LinkIt ONE development board’s flash storage
  - switch it into Mass Storage mode, as shown below



  - Navigate to **Computer**, then **Devices and Removable Storage**. Open the removable disk drive corresponding to the LinkIt ONE development board.

Provision your AWS certificates to the LinkIt ONE development board

- Copy the .pem files into the LinkIt ONE development board’s flash storage
  - Copy the certificates you saved in .pem format into the removable disk drive.
  - return the board to **Normal Bootup** switch mode (UART) and proceed with the sketch development.



Add the mbed TLS and AWS IoT Services MQTT libraries to the Arduino IDE

- AWS LinkIt ONE sketch development requires the mbed TLS and AWS IoT Services MQTT libraries to support MQTT communication
  - mbed TLS enables SSL/TLS capabilities for (embedded) products
  - find the source Arduino project (aws_paho_mqtt.ino) and the supporting libraries in the aws_mbedtls_mqtt repository on GitHub.
  - Copy the libraries into your Arduino/libraries folder.

Programming your application

- Open aws_paho_mqtt.ino sketch (from the aws_mbedtls_mqtt repository on GitHub)
- Resolve hostname for the IP address. Run the following command to identify the IP address,
  - **ping g.us-east-1.pb.iot.amazonaws.com**

  and Change the variable VMSTR IP_ADDRESS to the IP address you've just identified
- Provide the certificate names by matching the following variables
  - char cafileName[] = "A5.pem"; //root CA
  - char clientCRTName[] = "cert.pem";
  - char clientKeyName[] = "privatekey.pem";

Programming your application (cont.)

- Replace the topic name for the following variables
  - subParams.pTopic = "yourTopicName";
  - Params.pTopic = "yourTopicName";
- Upload the sketch to the LinkIt ONE development board.
  - Tools → Board → LinkIt One
  - Tools → Port
- Launch the Serial Monitor on the Arduino IDE
  - Tools → Port
  - Tools → Serial Monitor

Configure and Test Rules

Create an IAM Role for AWS IoT

- Assume Role policy

```
{
"Version": "2012-10-17",
"Statement": [{
    "Sid": "",
    "Effect": "Allow",
    "Principal": {
        "Service": "iot.amazonaws.com"
     },
    "Action": "sts:AssumeRole"
  }]
}
```

- **aws iam create-role --role-name "***iot-actions-role***" --assume-role-policy-document** file://*path-to-file/trust-policy-file*

Grant Permissions to the Role

- Create an IAM policy and attach the policy to your role

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

- **aws iam create-policy --policy-name "***iot-actions-policy***" --policy-document** file://*IAM-policy-document-file-path*
- **aws iam attach-role-policy --role-name "***iot-actions-role***" --policy-arn "***policy-ARN***"**

Create a Rule to Insert a Message into a DynamoDB Table

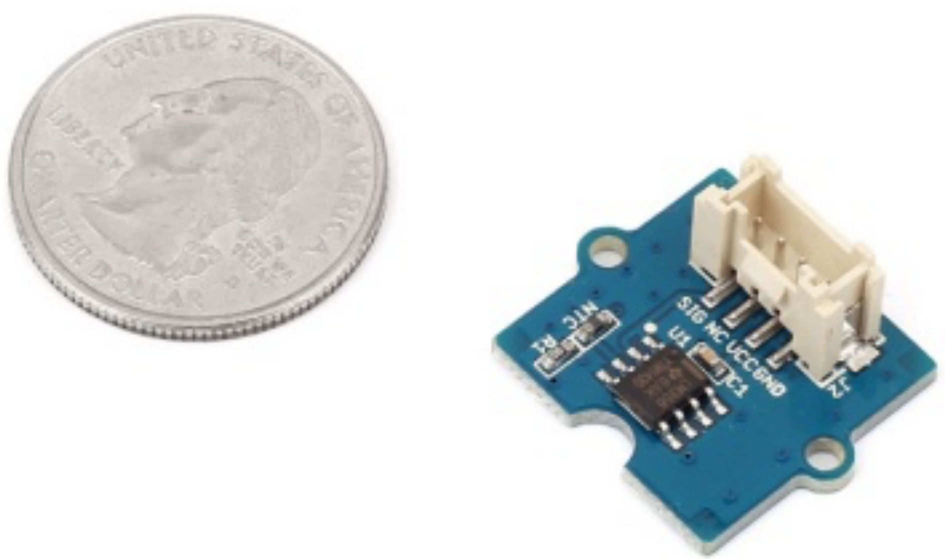
- Create a DynamoDB table has a hash attribute of type String and a range attribute of type String.
- Create an IoT Rule

```
{
 "sql": "SELECT * FROM 'topic/test'",
 "ruleDisabled": false,
 "actions": [{
   "dynamoDB": {
    "tableName": "sampleTable",
    "hashKeyField": "key",
    "hashKeyValue": "${topic(2)}",
    "rangeKeyField": "timestamp",
    "rangeKeyValue": "${timestamp()}",
    "roleArn": "arn:aws:iam::123456789012:role/iot-actions-role"
   }
  }]
}
```

- **aws iot create-topic-rule --rule-name "***saveToDynamoDB***" --topic-rule-payload** file://*path-to-file/DynamoDbRule*

Connect Grove Temperature Sensor

- Grove - Temperature Sensor V1.2
  - http://www.seeedstudio.com/wiki/Grove_-_Temperature_Sensor_V1.2



References

- http://docs.aws.amazon.com/iot/latest/developerguide/iot-quickstart.html
- https://labs.mediatek.com/site/global/developer_tools/mediatek_linkit/get-started/aws/get-started/index.gsp
- http://www.seeedstudio.com/wiki/Grove_-_Temperature_Sensor_V1.2
- http://www.slideshare.net/AmazonWebServices/mbl205-new-everything-you-want-to-know-about-aws-iot

Challenge – Win $100 USD AWS Credit !!

- 請使用 IoT Starter Kit 當中的 Moisture Sensor，建立一個自動記錄溼度的AWS IoT 解決方案，並將資料記錄在 DynamoDB中
- 完成之後請錄製一段短片 ，內容必須包括
  - 接上Moisture Sensor的LinkIt One開發版以及接收資料的DynamoDB.
  - 將Moisture Sensor插入水中，從DynamoDB可以收到變化的數值
- 將短片上傳到Youtube，標題為”我的 IoT 濕度計”，並將影片 連結email到 johnchan@amazon.com 驗證。
- [Hint]
  - https://github.com/Seeed-Studio/Moisture_Sensor