

TI and AWS showcase SimpleLink™ Wi-Fi and Amazon FreeRTOS: OTA Update Security and Reliability

Speakers:

Nick Lethaby, IoT Ecosystem Manager. Texas Instruments

Richard Barry, Principal Engineer, Amazon Web Services

Moderator:

Brandon Lewis, OpenSystems Media



Agenda

- 
- **Housekeeping**
 - **Presentation**
 - **Questions and Answers**
 - **Wrap-up**

Designing a Secure OTA Update Implementation

August, 2018

Nick Lethaby, IoT Ecosystem Manager, Texas Instruments

Richard Barry, Principal Engineer, AWS

Introduction

An OTA (Over-The-Air) update is a method of distributing and installing new firmware software updates via wireless connections

The ability to perform OTA updates is critical to IoT applications as it allows low-cost patching of bugs and security vulnerabilities, as well as addition of new features

In this webinar, we will examine

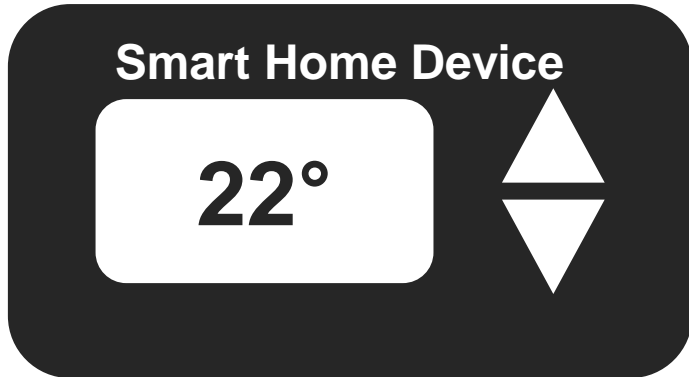
- 1 The potential for OTA updates to impact security and reliability
- 2 An OTA update solution based on an integration of Amazon FreeRTOS with the Texas Instruments SimpleLink Wi-Fi microcontrollers and how this addresses security and reliability concerns via:
 - Cloud-based services
 - Embedded software
 - Hardware architecture

OTA Updates: Understanding the risks



Reliability Risks:

- Flawed update crashes IoT device
- Power loss during update leaves non-functional image



Physical Attack:
Extract image or keys from memory

Physical Attack:
Boot malicious image



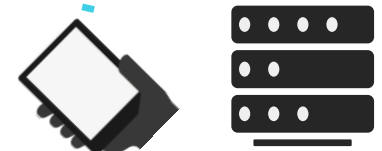
Physical Access

Remote Network Attack:
Send earlier version of image as OTA update

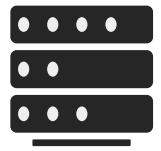
Local Network Attack:
Sniff packets



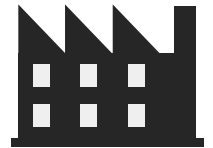
Remote Network Attack:
Unauthorized OTA operator



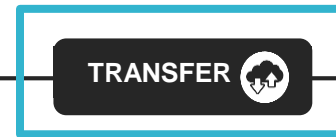
User



IoT Service



IoT Device Vendor



Local Network Access

Remote Access

AWS Cloud Computing



Compute



Storage



Database



Migration



Networking & Content Delivery



Developer Tools



Management Tools



Media Services



Security, Identity & Compliance



Analytics



Machine Learning



Mobile Services



AR & VR



Application Integration



Customer Engagement



Business Productivity



Desktop & App Streaming



Internet of Things



Game Development



AWS Cost Management

AWS Cloud Computing - OTA



Compute



Storage



Database



Migration



Networking & Content Delivery



Developer Tools



Management Tools



Media Services



Security, Identity & Compliance



Analytics



Machine Learning



Mobile Services



AR & VR



Application Integration



Customer Engagement



Business Productivity



Desktop & App Streaming



Internet of Things

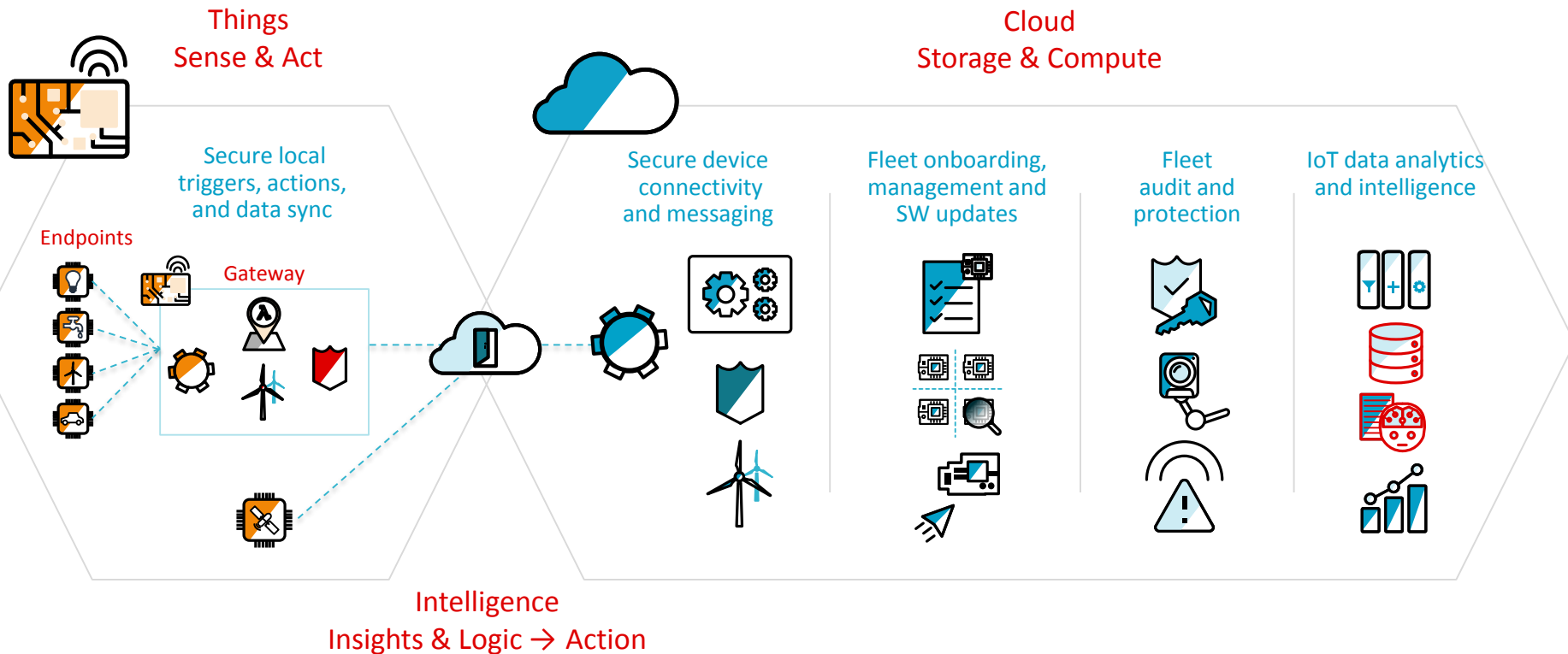


Game Development



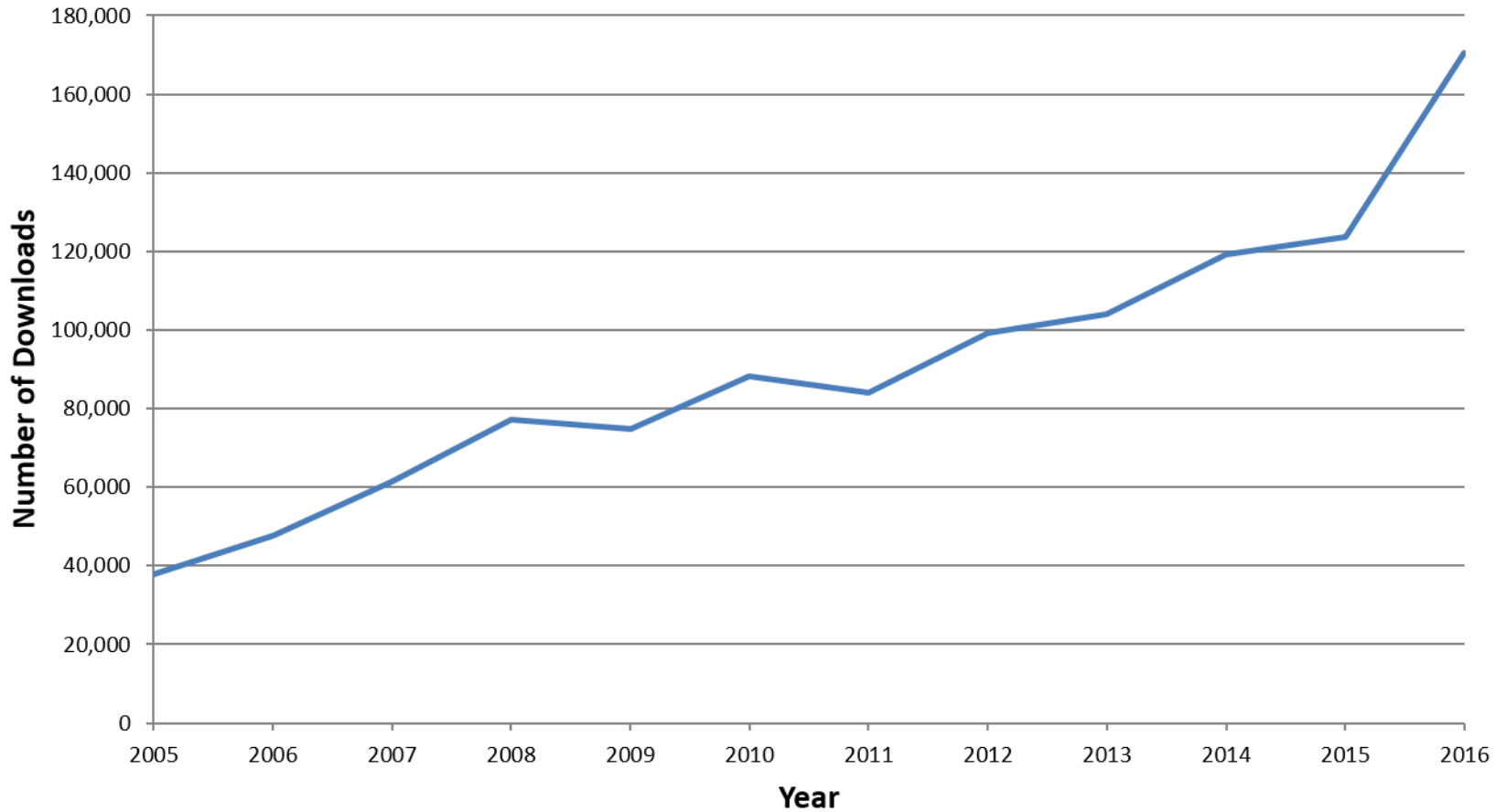
AWS Cost Management

AWS IoT Architecture



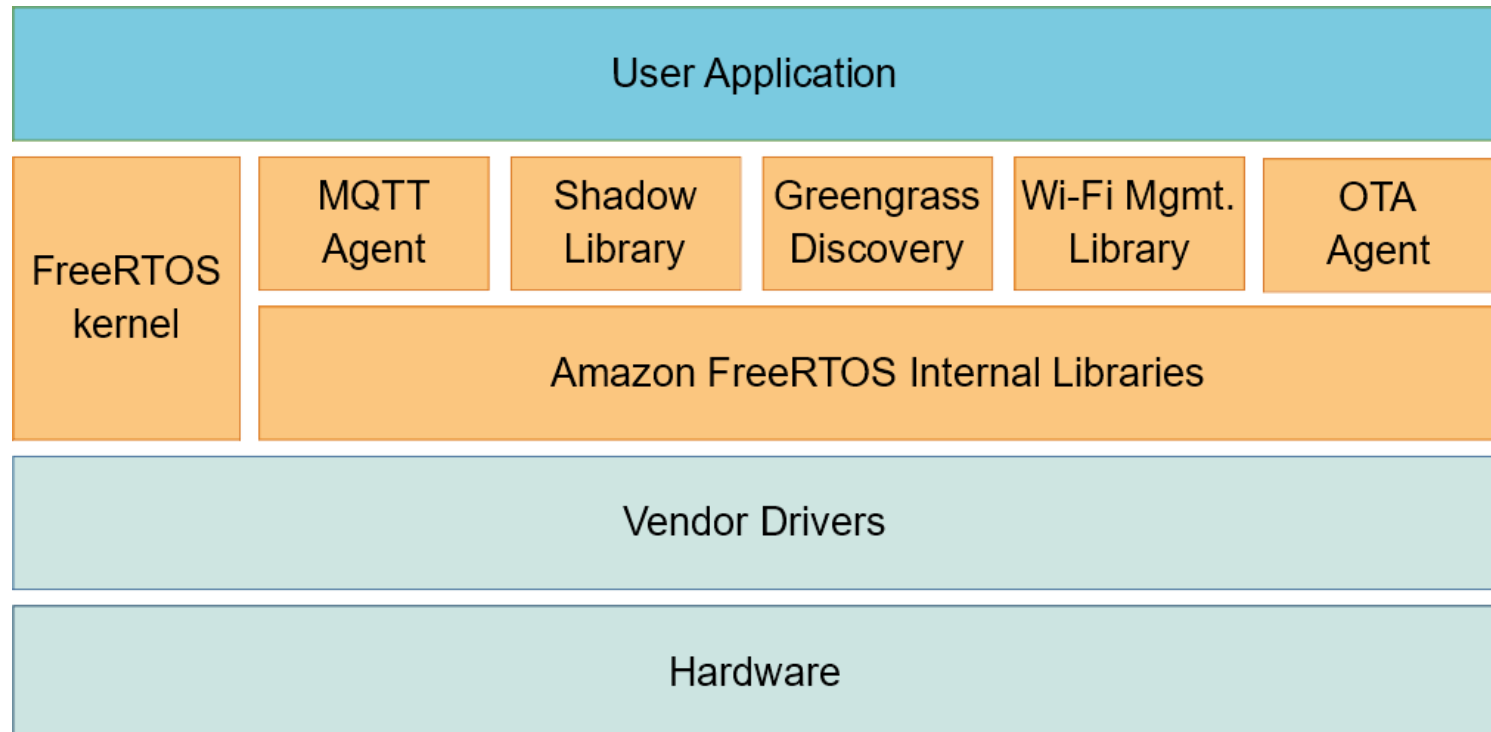
FreeRTOS Kernel

Downloads by Year

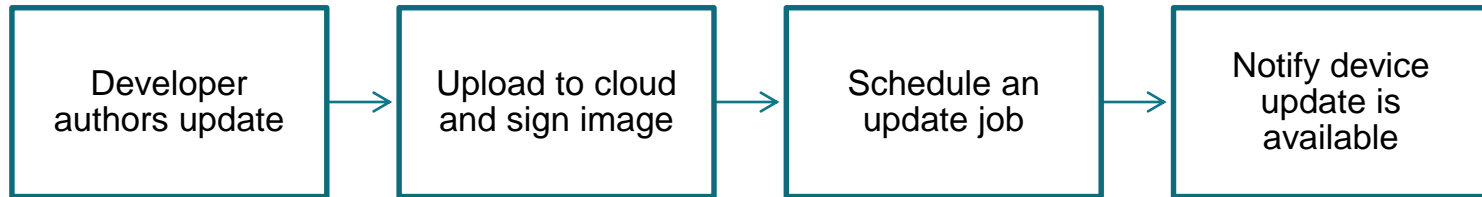


Amazon FreeRTOS

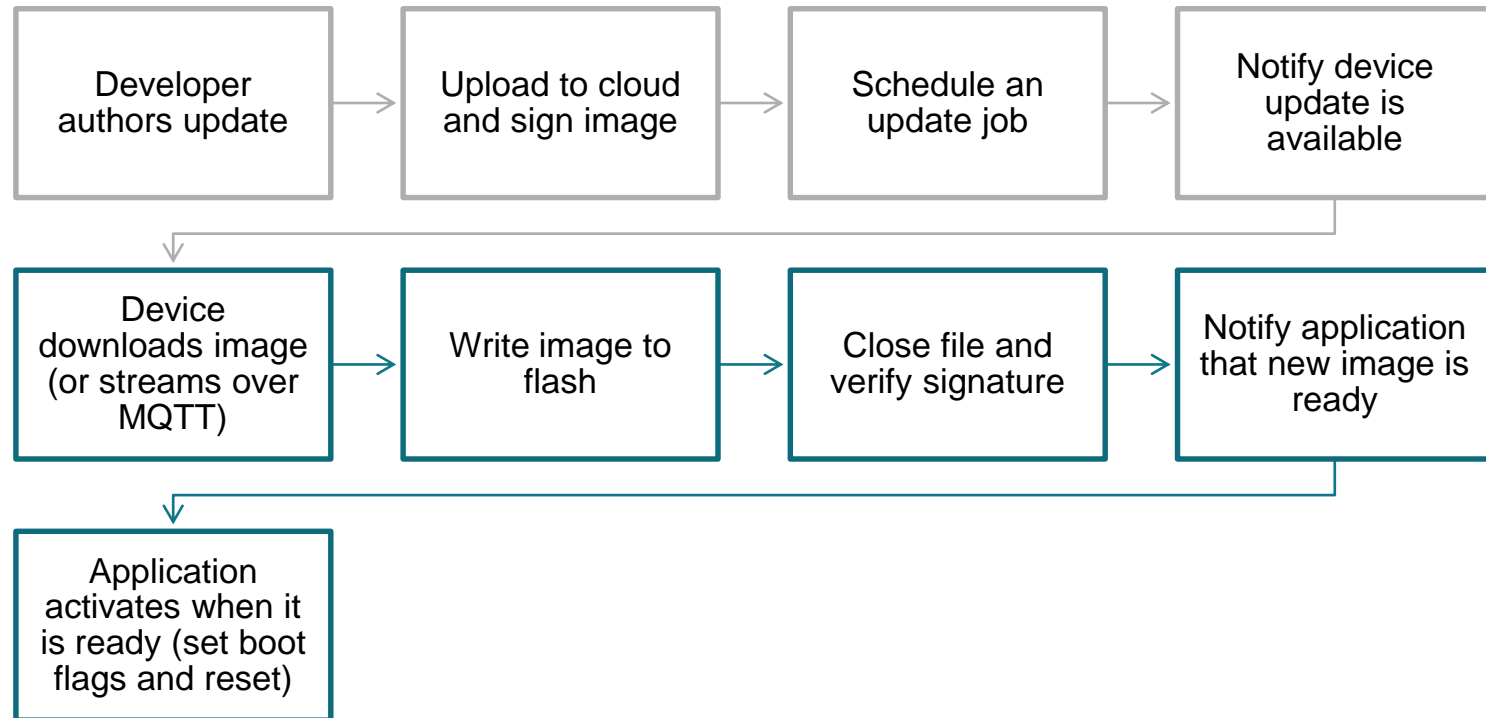
- Augments FreeRTOS kernel with functionality that enables MCUs to securely connect to cloud services
- Completely free to use for any application
- Open source MIT license



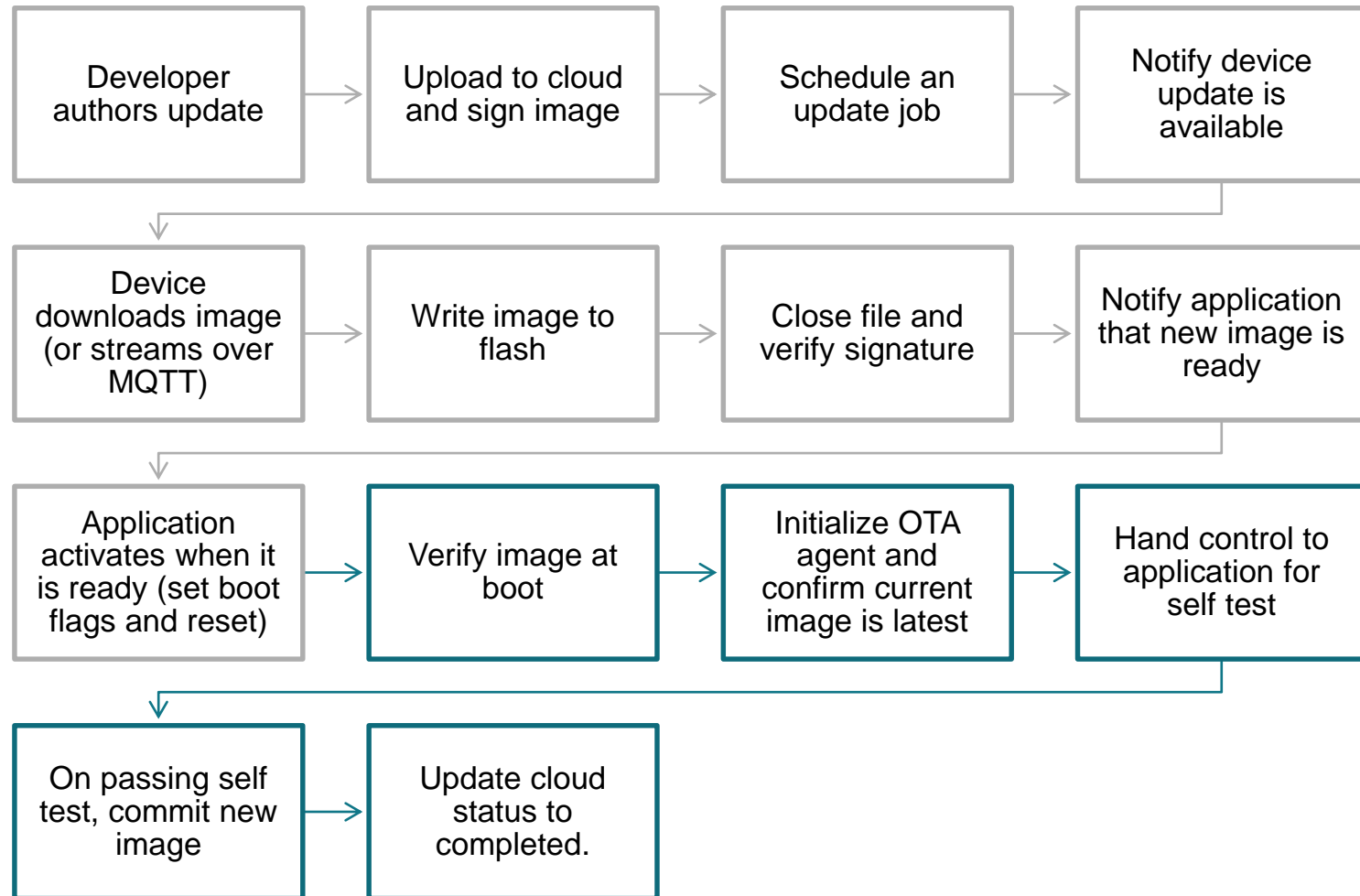
OTA on Amazon FreeRTOS: User Actions



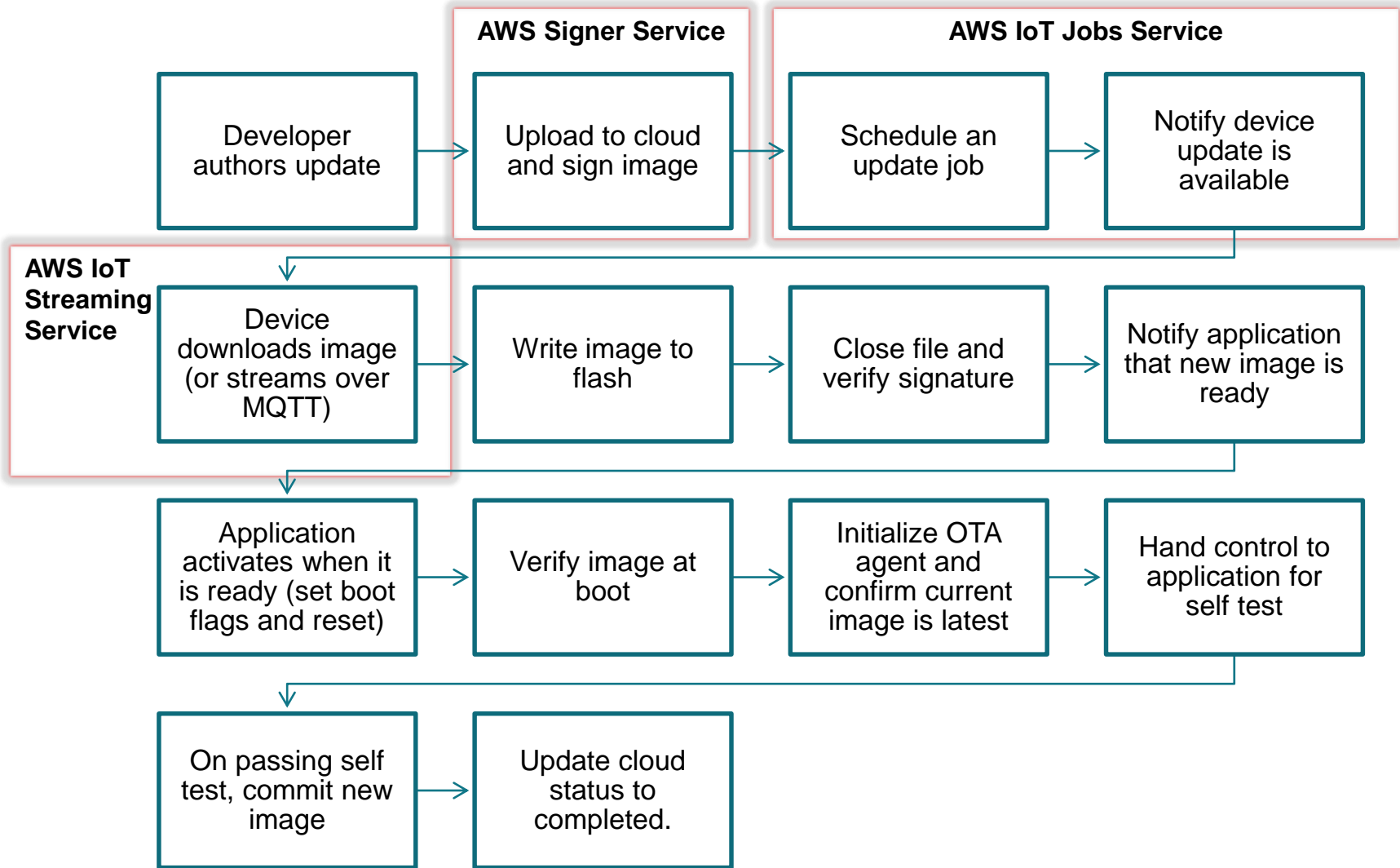
OTA on Amazon FreeRTOS: Agent Actions



OTA on Amazon FreeRTOS: Reboot Actions



OTA on Amazon FreeRTOS: An Overview



Code Signing Service

- Amazon FreeRTOS OTA updates require a signed image
- The IoT device can authenticate the source of the OTA image
- The signing service is integrated with the Amazon Certificate Manager (ACM)
- Device providers register their code signing certificate with the ACM

Create an Amazon FreeRTOS OTA update job BETA

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

Select devices to update
Browse and select the devices you want to include in this job.

No devices or thing groups selected [Select](#)

Select and sign your firmware image
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. [Learn more](#)

Sign a new firmware image for me
 Select a previously signed firmware image
 Use my custom signed firmware image

Device hardware platform

Pathname of code signing certificate on device [Learn more](#)

Pathname of firmware image on device [Learn more](#)

Select your firmware image in S3
 [Select](#)

Code signing certificate [Learn more](#)
 [Import a certificate](#) [Select](#)

Job Service (Scheduling Updates)

- OTA uses the AWS IoT Job Service
- The Job Service is used to define a set of remote operations (OTA is the operation in this case)
- You specify a list of targets to perform that job (a device group in this case)

Create an Amazon FreeRTOS OTA update job ^{BETA}

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

Select devices to update

Browse and select the devices you want to include in this job.

No devices or thing groups selected [Select](#)

Select and sign your firmware image

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. [Learn more](#)

Sign a new firmware image for me

Select a previously signed firmware image

Use my custom signed firmware image

Device hardware platform

CC3220SF-LAUNCHXL [Learn more](#)

Pathname of code signing certificate on device [Learn more](#)

e.g. /certificates/authcert.pem

Pathname of firmware image on device [Learn more](#)

/sys/mcuflashing.bin

Select your firmware image in S3

Image not selected [Select](#)

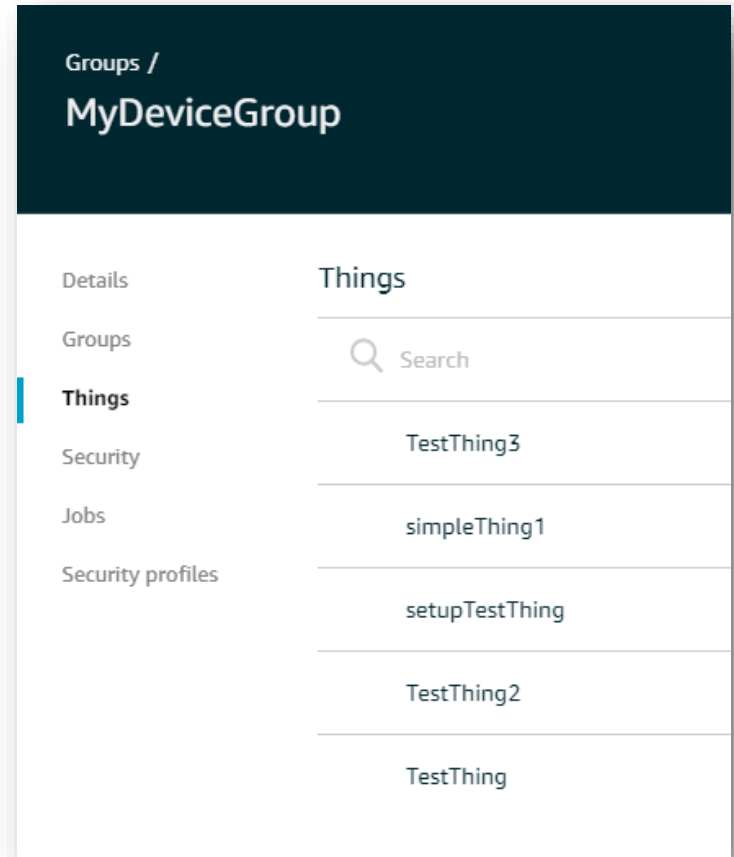
Code signing certificate [Learn more](#)

No certificate selected [Import a certificate](#) [Select](#)

Thing Groups

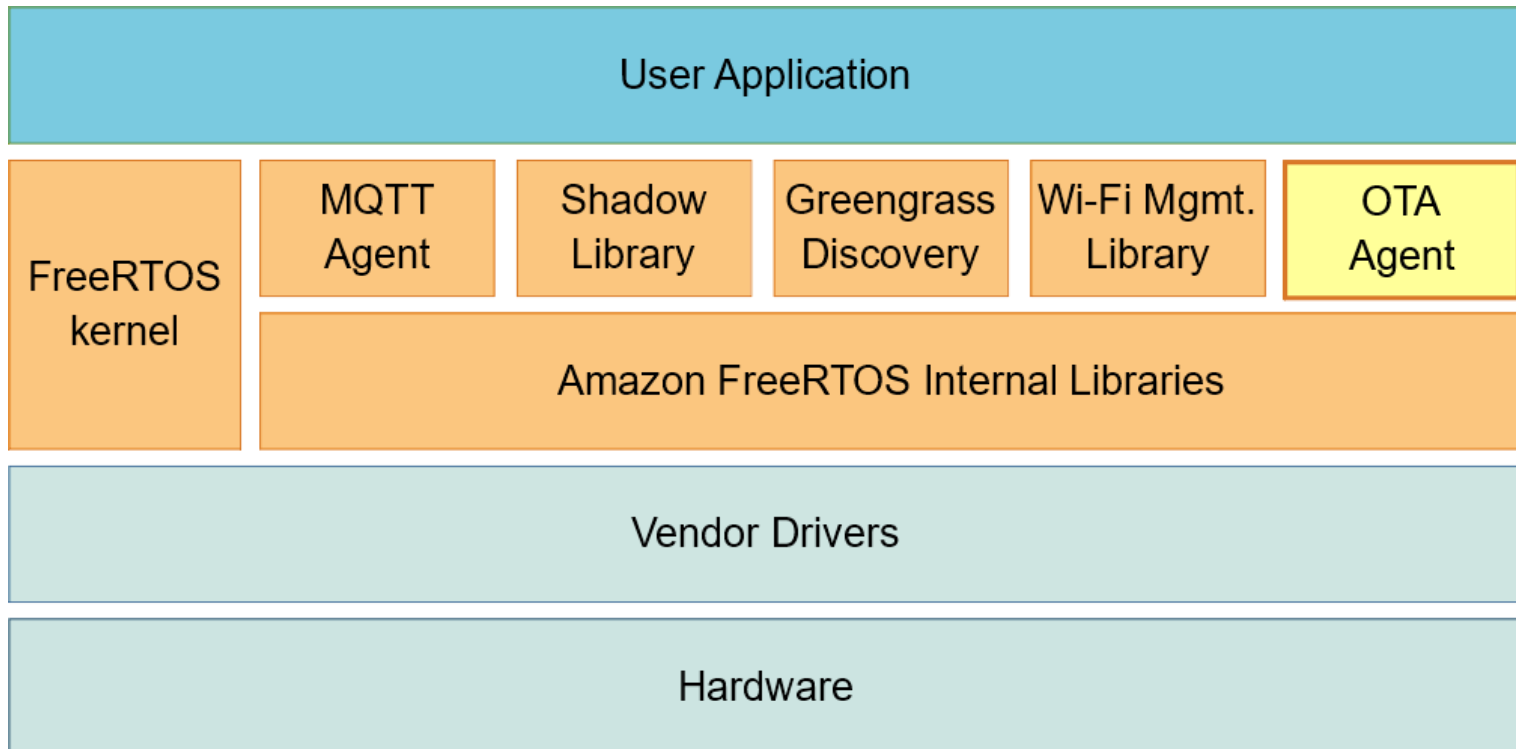
- Manage several devices/things at once by categorizing them into groups.
- Send OTA images to individual devices, or all devices in a group

```
{  "thingGroups": [  
    {  
      "groupName": "LightBulbs",  
      "groupArn": "arn:aws:iot:us-west-2:thinggroup/LightBulbs"  
    },  
    {  
      "groupName": "RedLights",  
      "groupArn": "arn:aws:iot:us-west-2:thinggroup/RedLights"  
    },  
  ]  
}
```



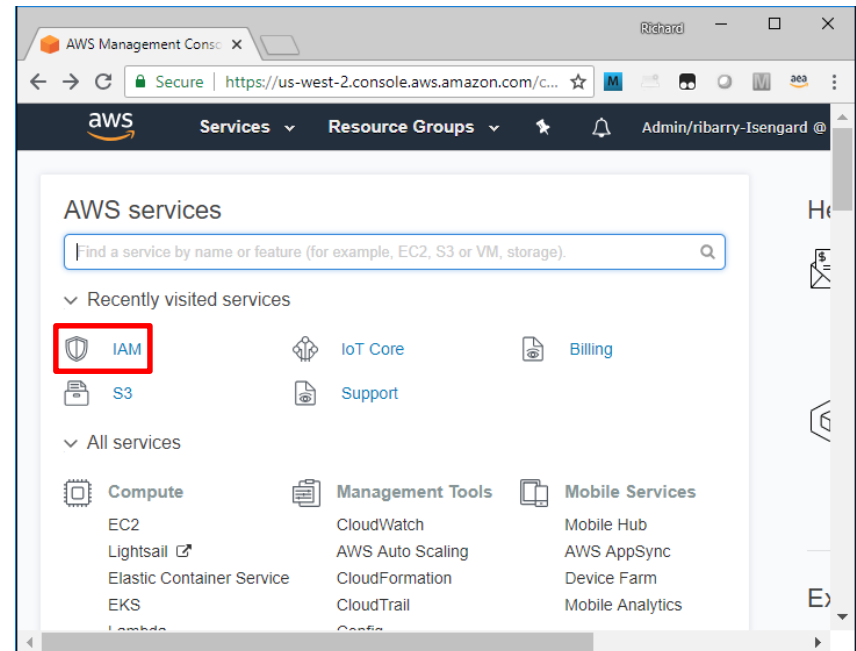
Device Side OTA Agent

- Uses MQTT Streaming Service and all communications through a single TLS connection
- Minimizes resource usage by downloading OTA via the existing TLS connection used for MQTT communication with AWS IoT



Cloud Operator Security Starts with IAM

- Identity and Access Management (IAM) lets you manage access to AWS services and resources securely
- Create and manage AWS users and groups
- Use permissions to allow and deny their access to AWS resources



IAM Users, Roles and Policies

- **IAM Users** allow you to define specific users of an AWS account with different permissions
- **IAM Roles** allow applications to access AWS services programmatically with specified permissions
- **IAM Policies** are documents that define the fine-grained permissions for each IAM User and Role

SAMPLE POLICY

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "FullAccess",
      "Effect": "Allow",
      "Action": ["s3:*"],
      "Resource": ["*"]
    },
    {
      "Sid": "DenyCustomerBucket",
      "Action": ["s3:*"],
      "Effect": "Deny",
      "Resource":
["arn:aws:s3:::customer",
"arn:aws:s3:::customer/*" ]
    }
  ]
}
```

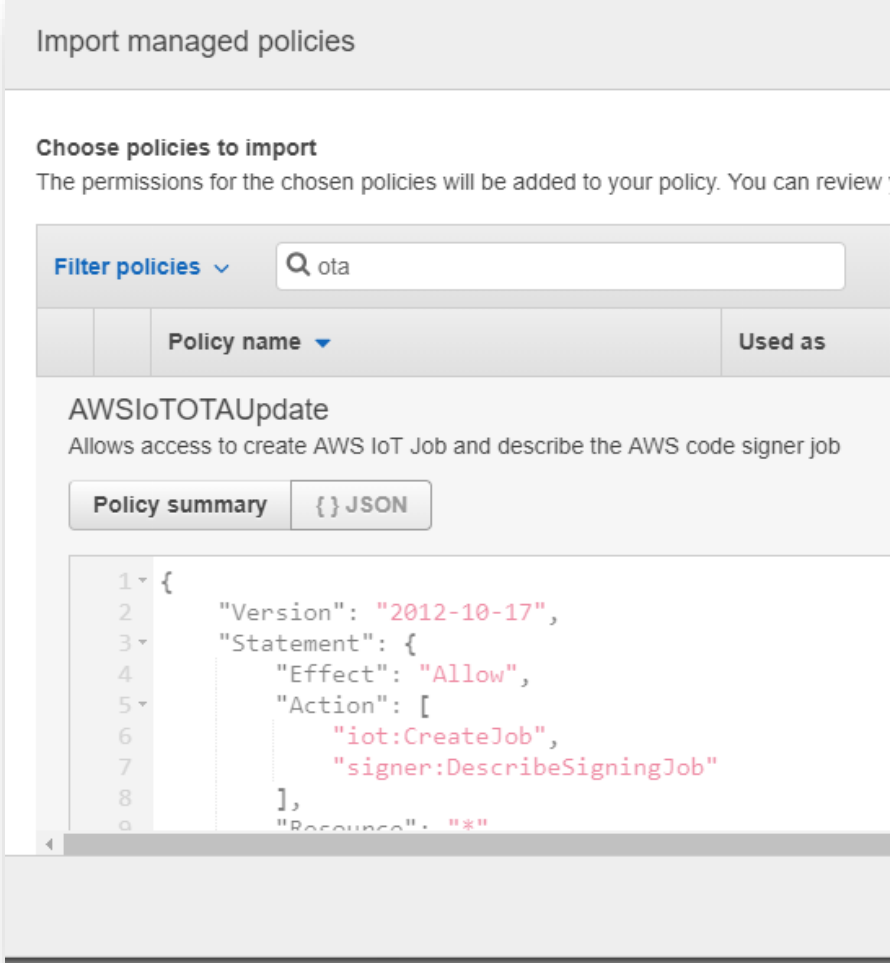
Preventing Unauthorized Job Operators

- An OTA User Policy grants your IAM user access to a number of job-related and OTA services.
- The following actions are performed during the FreeRTOS OTA workflow, and the following policies are therefore needed the IAM user.

```
"s3:ListBucket",  
"s3:ListAllMyBuckets",  
"s3:CreateBucket",  
"s3:PutBucketVersioning",  
"s3:GetBucketLocation",  
"s3:GetObjectVersion",  
  
"acm:ImportCertificate",  
"acm:ListCertificates",  
  
"iot:ListThings",  
"iot:ListThingGroups",  
"iot:CreateStream",  
  
"iot:CreateOTAUpdate",  
"iot:GetOTAUpdate",  
"iot:ListJobs",  
"iot:ListJobExecutionsForJob",  
,  
"iot:DescribeJob",  
"iot:GetJobDocument",  
"iam:ListRoles",  
  
"signer:ListSigningJobs",  
"signer:StartSigningJob",  
"signer:DescribeSigningJob"
```

Managed (Pre-Configured) Access Policies

- The OTA Service Role is a role that AWS IoT takes on to perform OTA actions on your behalf
- An AWS managed policy is a standalone policy that is created and administered by AWS, making it easier for you to assign appropriate permissions.
- Create a role, and add permissions to it using the managed policy called ***AWSIoTOTAUpdate***, which contains the permissions needed for AWS IoT to create jobs and use signed images.



Import managed policies

Choose policies to import
The permissions for the chosen policies will be added to your policy. You can review

Filter policies

Policy name	Used as
AWSIoTOTAUpdate	

AWSIoTOTAUpdate
Allows access to create AWS IoT Job and describe the AWS code signer job

Policy summary

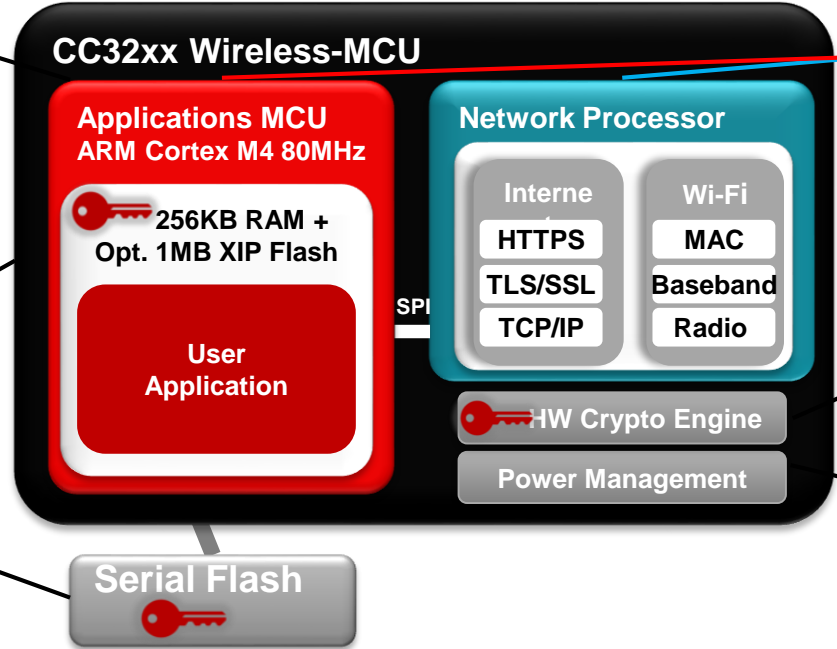
```
1 {
2   "Version": "2012-10-17",
3   "Statement": {
4     "Effect": "Allow",
5     "Action": [
6       "iot:CreateJob",
7       "signer:DescribeSigningJob"
8     ],
9     "Resource": "*"
10  }
```

SimpleLink Wi-Fi: Architected for better security

Single chip enclosed architecture for reduced attack surface

Embedded security features reduce need for external secure components

Encrypted File System for Customer IP/data and end user's data security



2 Separate execution environments: MCU + NWP for enhanced assets isolation and easy application integration

HW crypto engines enable fast TLS secure connection establishment within 200msec

Cryptographic utilities simplify sign & verify operations to validate any new image

Software

- File system security: Encryption, Access control, Authentication, Bundle protection, Software tamper detection, Cloning protection
- Initial secure programming
- Secure Boot

Embedded HW

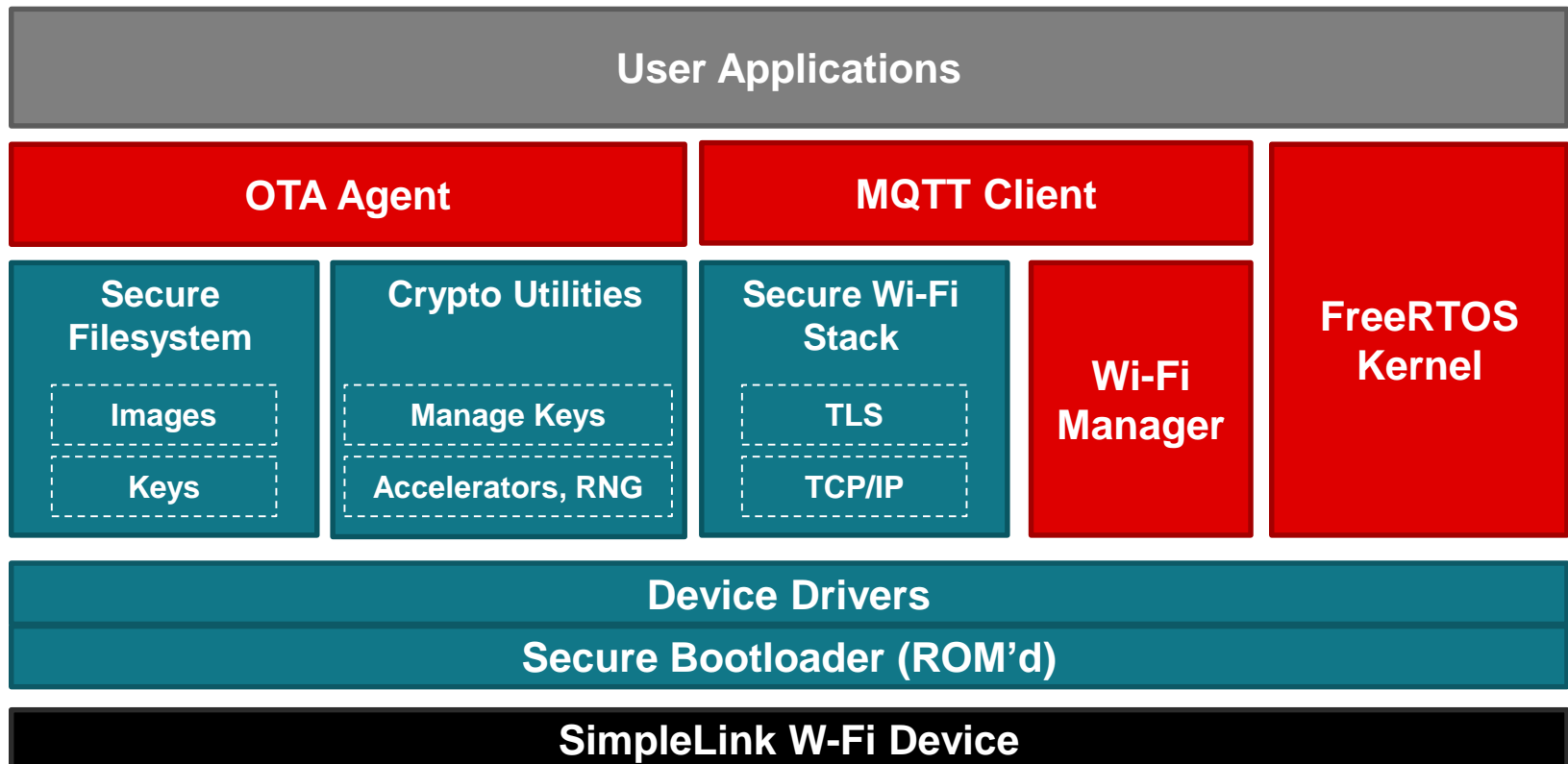
- Hardware Crypto Engine for advanced fast security, including: AES, DES, SHA/MD5, and CRC.
- Device-Unique Key
- Debug Security: JTAG and Debug Ports can be Locked

Networking

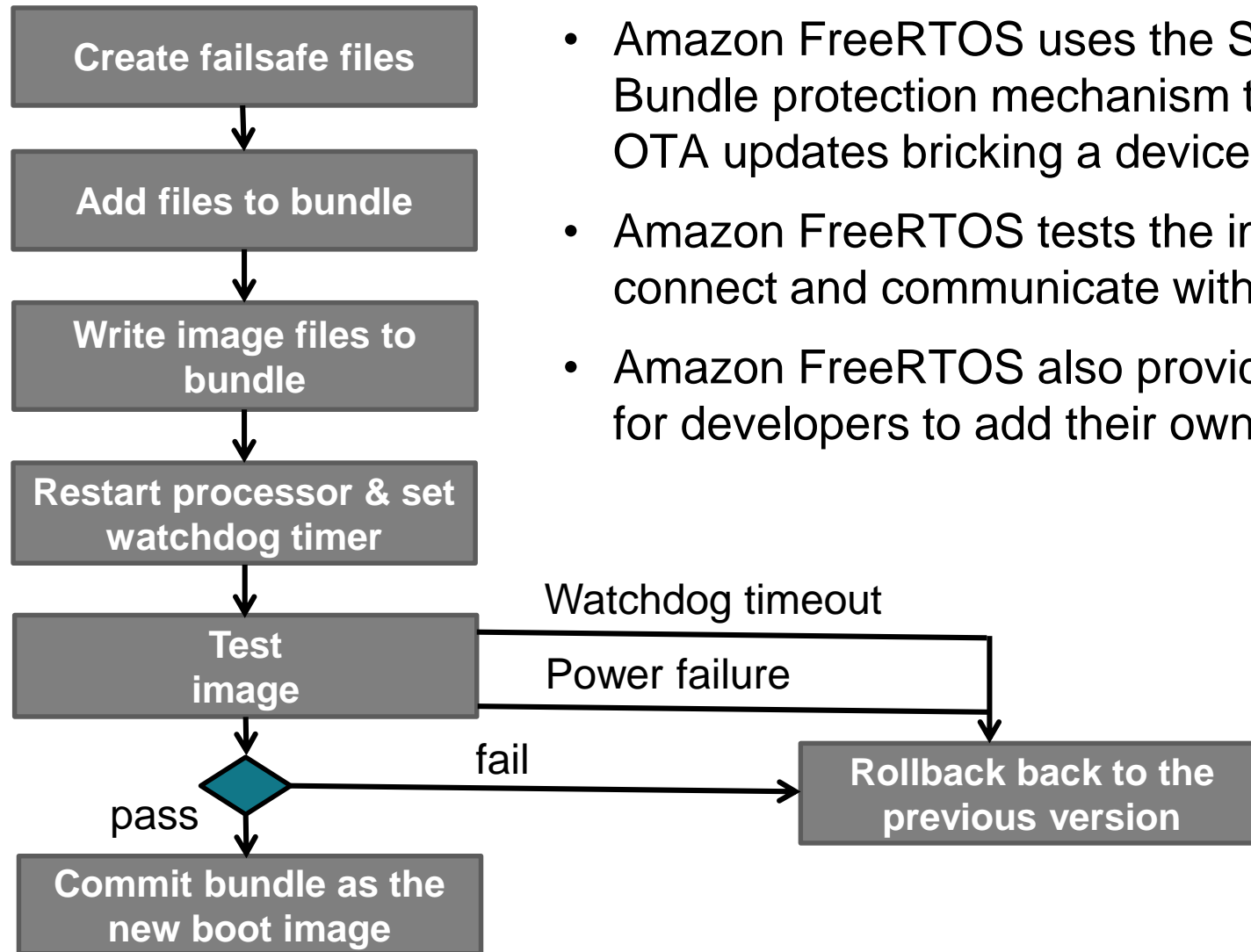
- Personal and enterprise security: WPA/WPA2 PSK, WPA2 Enterprise
- Full TCP/IP stack with TLS
- Embedded HTTPS Server
- Unique Device Identity
- Trusted Root-Certificate Catalog

Amazon FreeRTOS integration with SimpleLink SDK

- SimpleLink connected MCUs have a standard SDK across all devices
- The SimpleLink SDK feature extensive run-time libraries that Amazon FreeRTOS leverages in its secure OTA solution



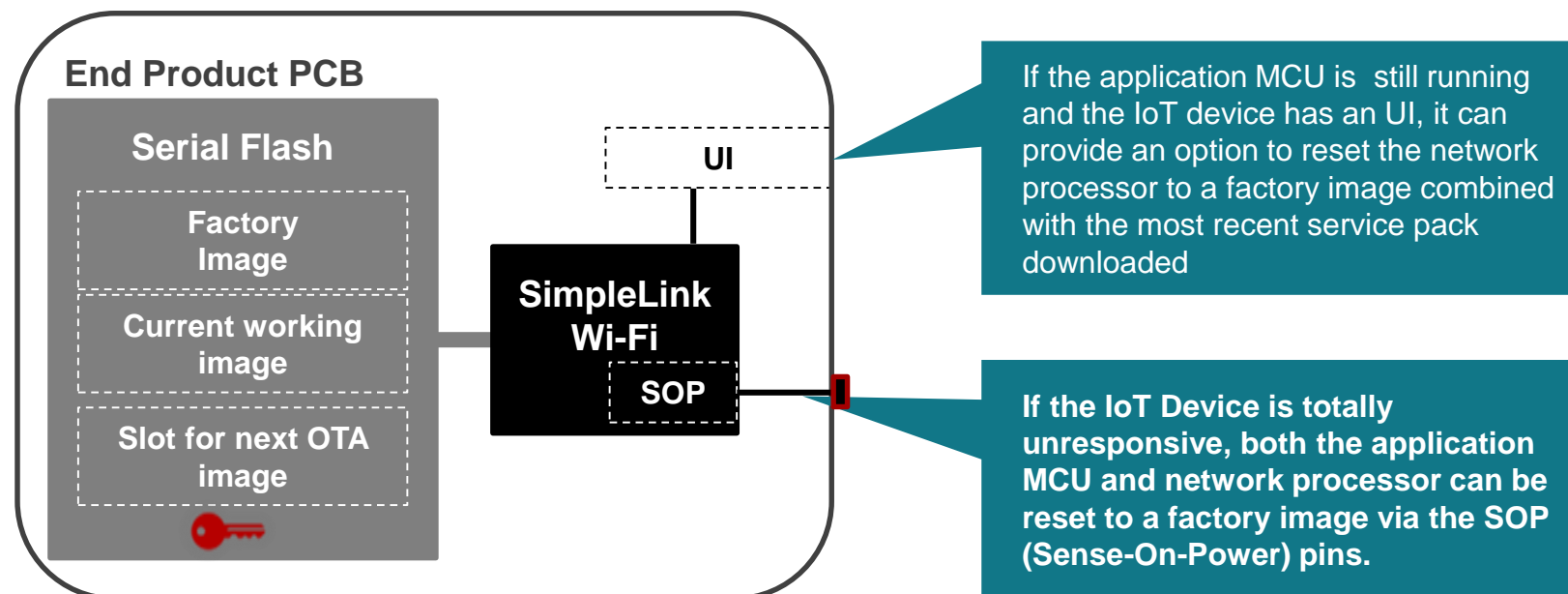
OTA reliability: SimpleLink Bundle Protection



- Amazon FreeRTOS uses the SimpleLink Bundle protection mechanism to avoid OTA updates bricking a device
- Amazon FreeRTOS tests the image can connect and communicate with AWS/IoT
- Amazon FreeRTOS also provides a hook for developers to add their own testing

OTA Reliability: Factory Image Recovery

- As an additional recovery mechanism to use for malfunctioning or bricked devices, SimpleLink Wi-Fi devices include a capability to restore to a device to a factory image



OTA Update Security Measures

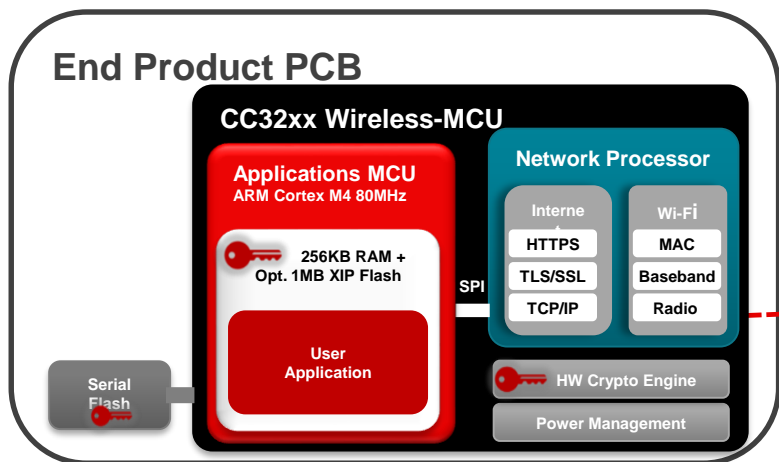
Hardware Crypto Engines

Local network with WPA/WPA2 encryption by the HW crypto engines, offloading the host

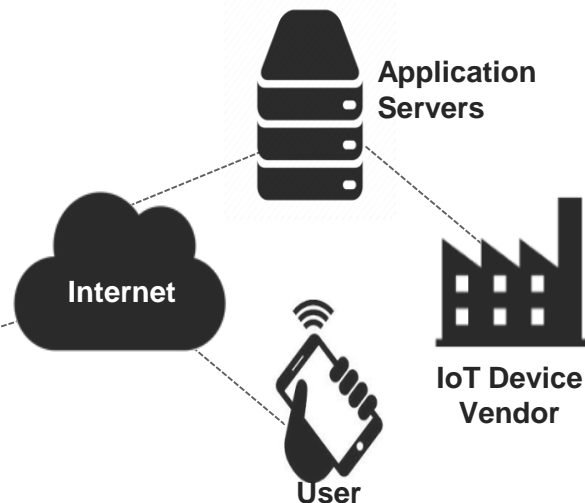
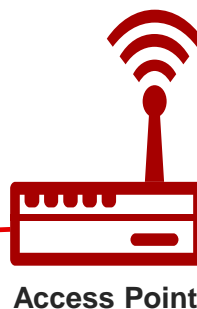
Personal and Enterprise Wi-Fi Security

OTA Security

The SimpleLink device opens a secured Wi-Fi connection to the Access-Point



Local Network Attack: Sniff packets



STORAGE



RUN-TIME



TRANSFER



Physical Access

Local Network Access

Remote Access

OTA Update Security Measures



Hardware Crypto Engines

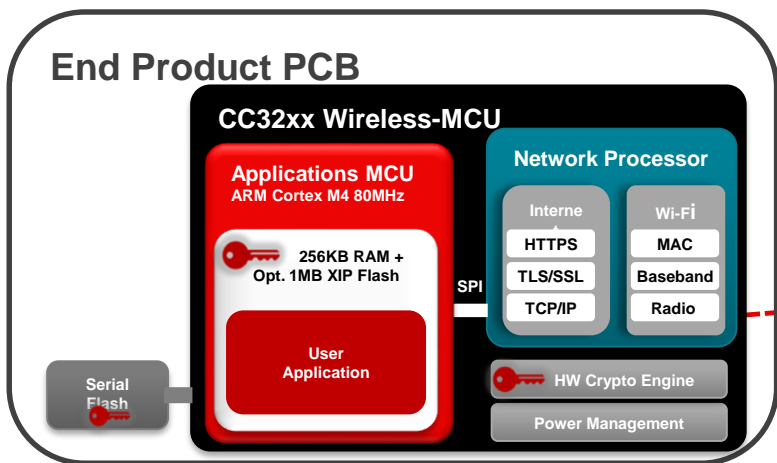
HW encryption engines establish a fast TLS/SSL internet connection within <200mSec

Secure Sockets (TLS/SSL)

TLS/SSL are in the NWP, within the BSD Socket layer

OTA Security

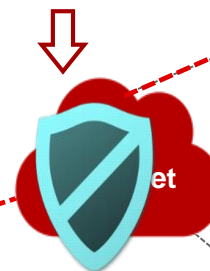
The SimpleLink device has a secured TLS connection to the AWS IoT service.



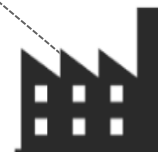
Remote Attack:
Sniff packets



Access Point



Application Servers



IoT Device Vendor



User

STORAGE



RUN-TIME



Physical Access

TRANSFER



Local Network Access

Remote Access



TEXAS INSTRUMENTS

OTA Update Security Measures

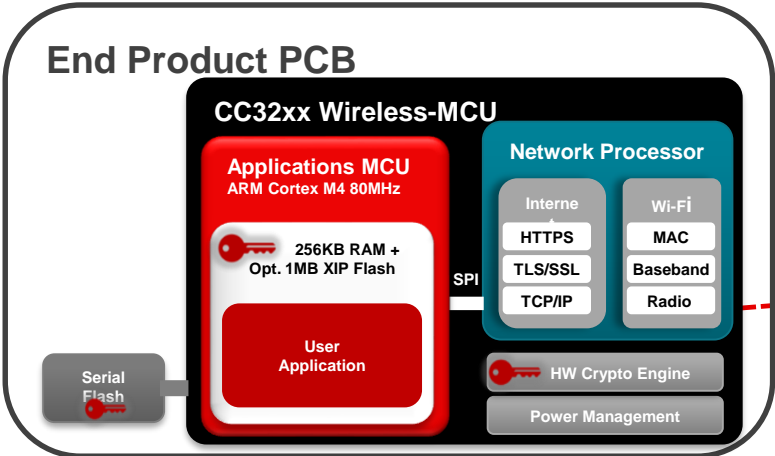
Hardware Crypto Engines

Trusted Root-Certificate Catalog

Built-in secure mechanism to ensure a CA is trusted as root of certificate chain for TLS purpose and file signing.

TI Root of Trust Public Key

HW-based mechanism that allows authenticating TI as the genuine origin of a given content, using asymmetric keys.



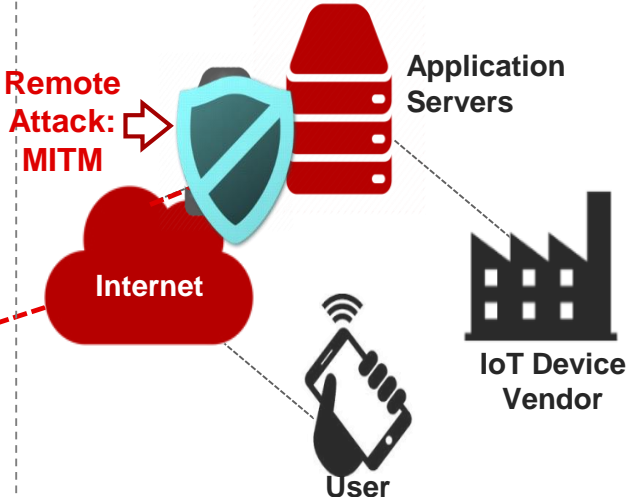
STORAGE

RUN-TIME

Physical Access



Local Network Access



TRANSFER

Remote Access

OTA Security
During the TLS connection the server is authenticated to the SimpleLink's trusted root certificate catalog.

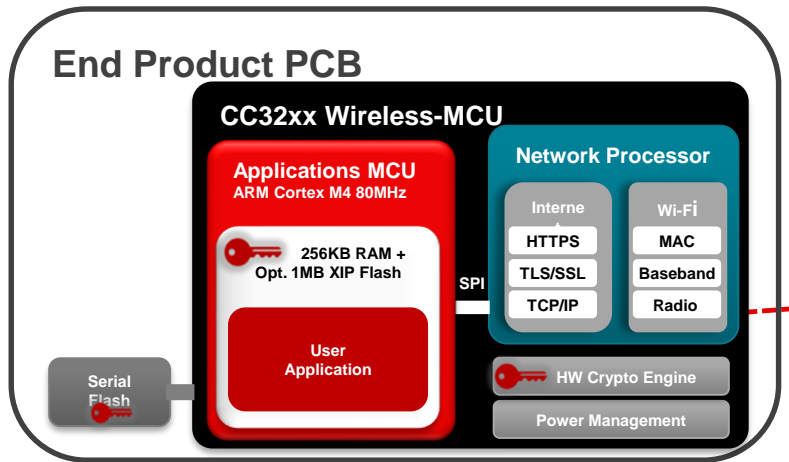
OTA Update Security Measures

Hardware Crypto Engines

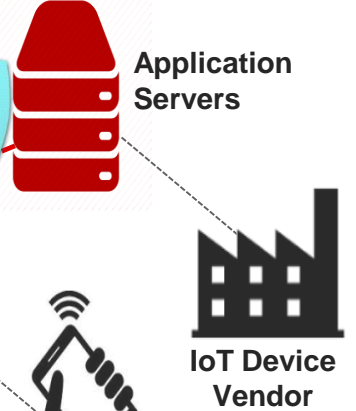
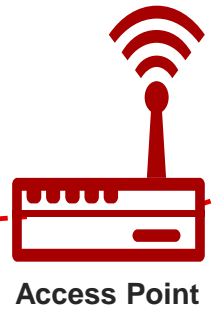
Device Identity

Unmodifiable unique 128-bit number that TI burns into the device during production

OTA Security
The simplelink device uses its unique device identity in order to be validated and approved to continue with a SW update



Unauthorized device asking for the update



Physical Access

Local Network Access

Remote Access

OTA Update Security Measures

Hardware Crypto Engines

File System Security

Unique Key - Cloning Protection

File Encryption

File system is encrypted so image cannot be read without the key

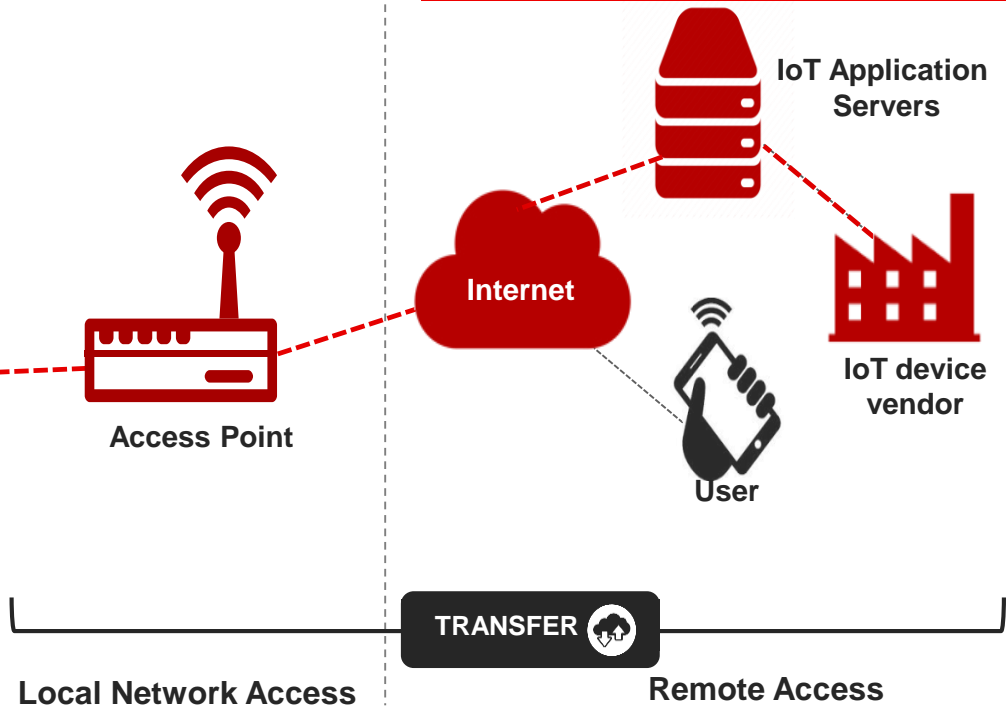
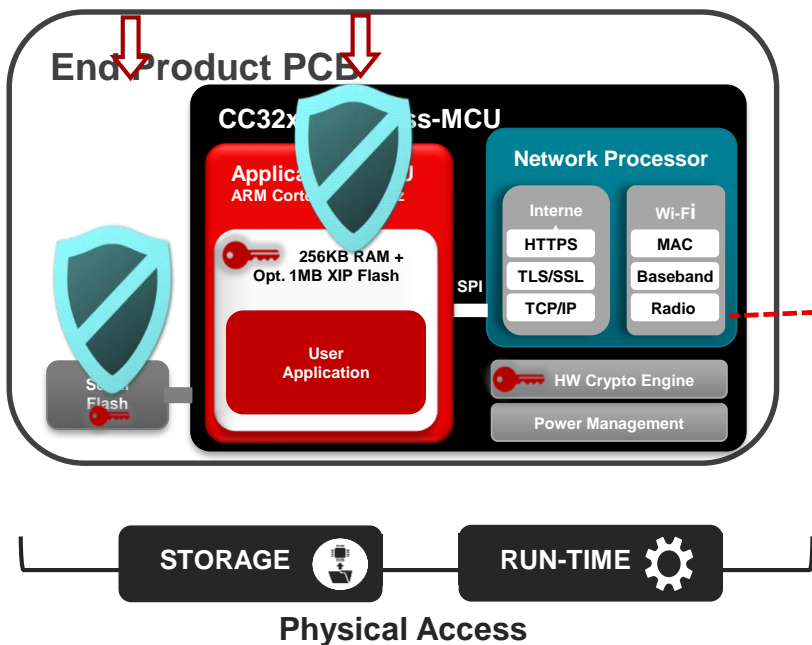
The file system is readable only by the device which first booted the image

Physical Access Attack:
Read App Code or keys

Physical Access Attack:
Copy and run counterfeit HW

OTA Security

The updated files access control and authenticity are validated and then stored on the Simplelink's secured file system



OTA Update Security Measures

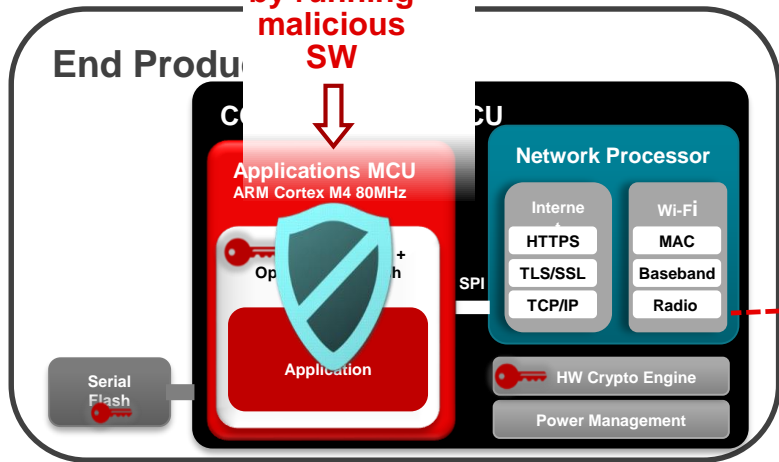


Hardware Crypto Engines

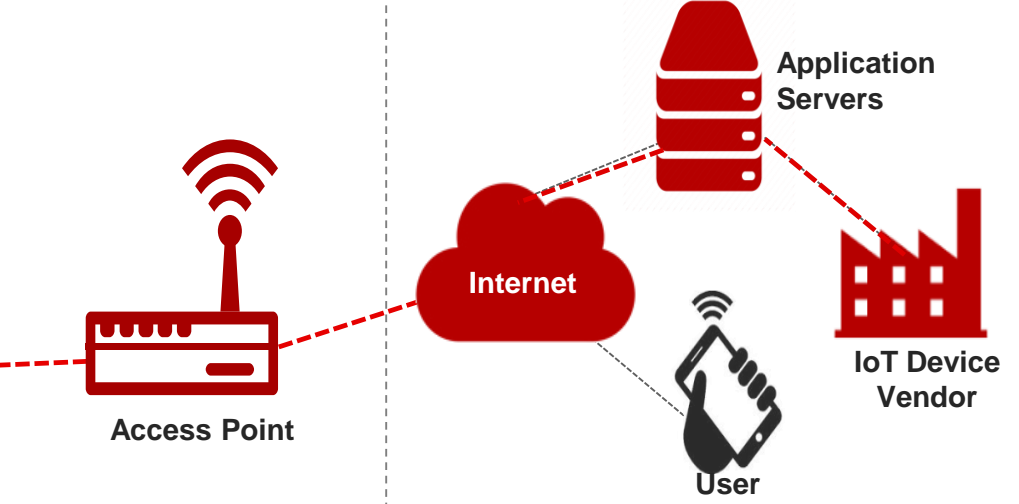
Validate the integrity and authenticity of the run-time binary during boot

Secure Boot

Attempt to hijack device by running malicious SW



OTA Security
At boot the application code is loaded and run on the MCU



STORAGE

RUN-TIME

TRANSFER

Physical Access

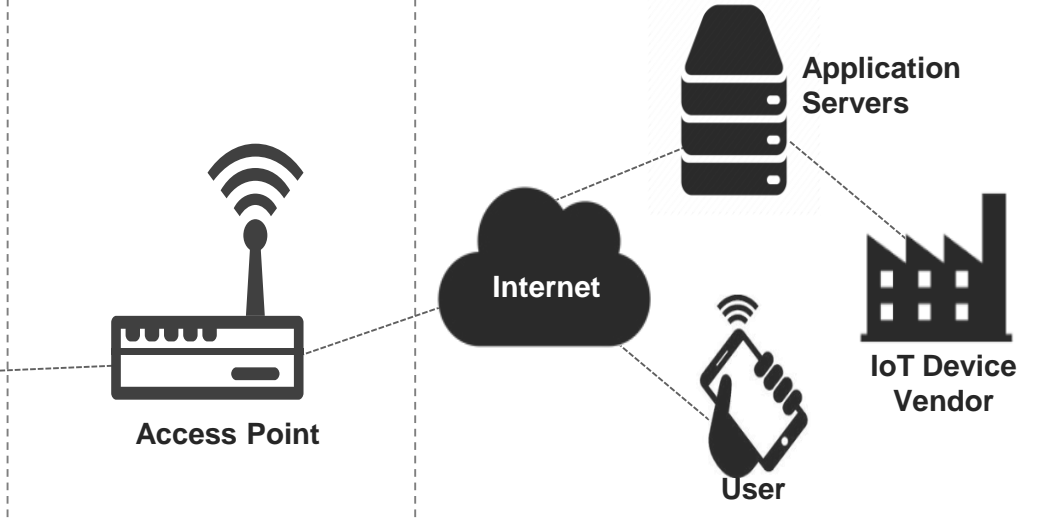
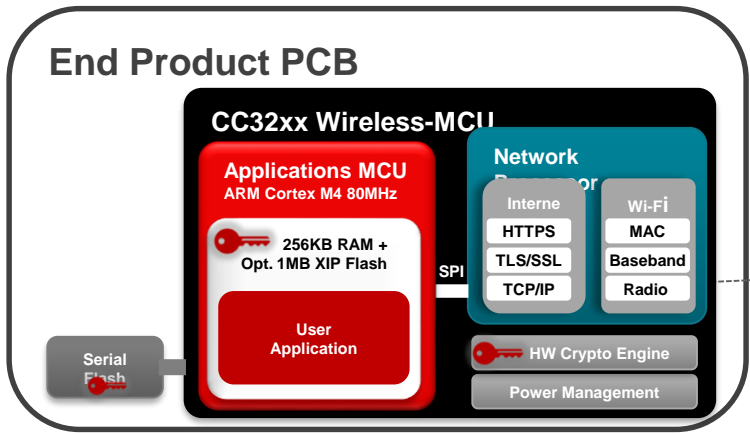
Local Network Access

Remote Access



SimpleLink Wi-Fi: Multi Layer Security Measures

	Hardware Crypto Engines	
	Trusted Root-Certificate Catalog	
File System Security	Debug Security	Secure Sockets (TLS/SSL)
	Unique Key - Cloning Protection	Device Identity
	Software Tamper Detection	Secure Key Storage
	File Encryption	Secure Content Delivery
	File Authentication	
	File Access Control	Personal and Enterprise Wi-Fi Security
	Factory Image Recovery	HTTPS Service
	File Bundle Protection	
Secure Boot		
Initial Secure Programming		



For more information

SimpleLink Wi-Fi devices and tools

- <http://www.ti.com/wireless-connectivity/simplelink-solutions/wi-fi/overview/overview.html>

Getting started with a CC3220SF Launchpad

- <http://www.ti.com/tool/CC3220SF-LAUNCHXL>

Review CC3220 technical documents

- <http://www.ti.com/product/CC3220/technicaldocuments>

Amazon FreeRTOS and Secure OTA

- <https://aws.amazon.com/freertos/>
- <https://docs.aws.amazon.com/freertos/latest/userguide/freertos-ota-dev.html>

AWS IoT

- <https://aws.amazon.com/iot/>

Summary

OTA Updates are a critical capability for an IoT device but introduce the potential for security and reliability risks



Amazon offers an end-to-end secure OTA solution based on AWS IoT cloud services and Amazon FreeRTOS embedded software



Amazon FreeRTOS is integrated with the SimpleLink SDK, enabling it to leverage SimpleLink Wi-Fi's built-in OTA security and reliability features



Audience Q & A

Nick Lethaby,
IoT Ecosystem Manager.
Texas Instruments

Richard Barry,
Principal Engineer,
Amazon Web Services



Thanks for joining us



Event archive available at:

<http://ecast.opensystemsmedia.com/>

E-mail us at: jgilmore@opensystemsmedia.com