AWSOTAUG

Amazon FreeRTOS Over-The-Air Updates using i.MX RT1060

Rev. 2 — 15 June 2022 User guide

Document information

Information	Content
Keywords	OTAP, AWS, AWS services, Amazon FreeRTOS, Over The Air, RT1060-EVK SDK
Abstract	This document lists the steps to configure AWS services to make an Amazon FreeRTOS Over The Air Update using NXPs RT1060-EVK SDK

1 Overview

This guide walks through the steps to configure AWS services to make an Amazon FreeRTOS Over The Air Update using NXPs RT1060-EVK SDK. First, it creates an IAM role with OTA update, S3, IoT policies, and permissions. Then, using OpenSSL and AWS CLI commands, a code signing certificate is issued. Finally, it shows how to create an IoT thing with the code signing certificate with an OTA job.

Note: *The figures used in the document might be slightly different due to background changes by Amazon.*

2 AWS OTA prerequisites

2.1 Create an Amazon S3 bucket and store your update

1. Go to the https://console.aws.amazon.com/s3/.
2. Choose **Create bucket**.

3. Type a bucket name, and then choose **Next**.
4. Select **Versioning** to keep all versions in the same bucket, and then choose **Next**.

5. Choose **Next** to accept the default permissions.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

6. Choose **Create bucket**.

2.2 Create an OTA update service role

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2.2.1 Create an OTA service role

1. Sign in to the https://console.aws.amazon.com/iam/.
2. From the navigation pane, choose **Roles**.

3. Choose to **Create role**.
4. Under **Select type of trusted entity**, choose **AWS Service**.

5. Choose **IoT** from the list of AWS services.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

6. Under **Select your use case**, choose **IoT**.

7. Choose **Next: Permissions**.

8. Choose **Next: Tags**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

9. Choose **Next: Review**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

10. Enter a role name and description and then choose to **Create role**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2.2.2 To add OTA update permissions to your OTA service role

1. In the search box on the IAM console page, enter the name of your role, and then choose it from the list.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. Choose **Attach policies**.

3. In the **Search** box, enter **AmazonFreeRTOSOTAUpdate**, select **AmazonFreeRTOSOTAUpdate**.
4. From the list of filtered policies, and then choose **Attach policy** to attach the policy to your service role.

2.2.3 To add the required IAM permissions to your OTA service role

1. Choose **Add inline policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. Choose the **JSON** tab.
3. Copy and paste the following policy document into the text box:

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "iam:GetRole",
            "iam:PassRole"
         ],
         "Resource": "arn:aws:iam::<your_account_id>:role/
<your_role_name>"
      }
    ]
}
```

Make sure that you replace *<your_account_id>* with your AWS account ID, and *<your_role_name>* with the name of the OTA service role.
Note: *To obtain account ID, select account name in Web page menu bar and select* ***My account*** *from the drop-down menu. Make note of the* ***Account ID*** *under* ***Account Settings****.*

4. Choose **Review policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

5. Enter a name for the policy, and then choose **Create policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2.2.4 To add the required Amazon S3 permissions to your OTA service role

1. In the search box on the IAM console page, enter the name of your role, and then choose it from the list.
2. Choose **Add inline policy**.

3. Choose the **JSON** tab.
   Copy and paste the following policy document into the box:

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "s3:ListBucketVersions",
            "s3:GetObjectVersion",
       "s3:GetObject",
       "s3:PutObject"
         ],
         "Resource": [
        "arn:aws:s3:::<example-bucket>/*"
        "arn:aws:s3:::<example-bucket>"
     ]
      }
     ]
}
```

   This policy grants your OTA service role permission to read Amazon S3 objects. Make sure that you replace *<example-bucket>* with the name of your bucket.
4. Choose **Review policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

5. Enter a name for the policy, and then choose **Create policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2.3 Create an OTA user policy

1. Open the https://console.aws.amazon.com/iam/ console.
2. In the navigation pane, choose **Users**.
3. Choose your IAM user from the list.
4. Choose **Add permissions**.

5. Choose **Attach existing policies directly**.

6. Choose **Create policy**.

Choose the **JSON** tab, and copy and paste the following policy document into the policy editor:

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "s3:ListBucket",
      "s3:ListAllMyBuckets",
      "s3:CreateBucket",
      "s3:PutBucketVersioning",
      "s3:GetBucketLocation",
            "s3:GetObjectVersion",
      "acm:ImportCertificate",
      "acm:ListCertificates",
      "iot:*",
      "iam:ListRoles",
      "freertos:ListHardwarePlatforms",
```


All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

```
        "freertos:DescribeHardwarePlatform"
            ],
            "Resource": "*"
        }
    {
            "Effect": "Allow",
            "Action": [
        "s3:GetObject",
        "s3:PutObject"
            ],
            "Resource": "arn:aws:s3:::<example-bucket>/*"
        }
    {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::<your-account-id>:role/
<role-name>"
        }
    ]
}
```

Replace *<example-bucket>* with the name of the Amazon S3 bucket where your OTA update firmware image is stored. Replace *<your-account-id>* with your AWS account ID. You can find your AWS account ID in the upper right of the console. When you enter your account ID, remove any dashes (-). Replace *<role-name>* with the name of the IAM service role that you created.

1. Choose **Review policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. Enter a name for your new OTA user policy, and then choose **Create policy**.

2.4 Windows prerequisites

2.4.1 OpenSSL

1. Install OpenSSL

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. Modify the system environment variable path and add your OpenSSL bin directory

Make sure that openssl gets assigned to the OpenSSL executable in your command prompt or terminal environment.

2.4.2 Install the AWS CLI

1. Follow the instructions for AWS CLI bundler installer https://docs.aws.amazon.com/cli/latest/userguide/install-windows.html#install-msi-on-windows
2. Go to the IAM console https://console.aws.amazon.com/iam/

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

3. In the navigation pane, choose **Users**.

4. Choose your IAM user account.
5. Select **Security credentials**

6. In the **Access keys** section, choose **Create access key**.

7. To view the new access key pair, choose **Show**. You will not have access to the secret access key again after this dialog box closes. Your credentials look something like this: *Access key ID: AKIAIOSFODNN7EXAMPLE Secret access key: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY*
8. To download the key pair, choose **Download .csv** file. Store the keys in a secure location. You will not have access to the secret access key again after this dialog box closes. Keep the keys confidential to protect your AWS account and never email

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

them. Do not share them outside your organization, even if an inquiry appears to come from AWS or Amazon.com. No one who legitimately represents Amazon asks you for your secret key.

9. After you download the .csv file, choose **Close**. When you create an access key, the key pair is active by default, and you can use the pair right away.
10. For general use, the aws configure command is the fastest way to set up your AWS CLI installation

2.5 Creating a code-signing certificate

1. In your working directory, use the following text to create a file named cert_config.txt. Replace *test_signer@amazon.com* with your email address: [req]

```
prompt = no
distinguished_name = my_dn
[ my_dn ]
commonName = test_signer@amazon.com
[ my_exts ]
keyUsage = digitalSignature
extendedKeyUsage = codesigning
```

2. Using openSSL command line create an ECDSA code-signing private key: openssl genpkey-algorithm EC -pkeyopt ec_paramgen_curve:P-256 -pkeyopt ec_param_enc:named_curve -outform PEM -out ecdsasigner.key

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

3. Create an ECDSA code-signing certificate: openssl req -new -x509 -config cert_config.txt -extensions my_exts -nodes -days 365 -key ecdsasigner.key -out ecdsasigner.crt
4. Import the code-signing certificate, private key, and certificate chain into AWS Certificate Manager: aws acm import-certificate *--certificate file://ecdsasigner.crt --private-key file://ecdsasigner.key* ***Note:*** *this command displays an ARN for your certificate. Save it locally to use it while creating the OTA update job*.

3 Grant access to code signing for AWS IoT

1. Sign in to the https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Policies**.

3. Choose **Create Policy**.

4. On the **JSON** tab, copy and paste the following JSON document into the policy editor. This policy allows the IAM user access to all code-signing operations.

```
{
   "Version": "2012-10-17",
   "Statement": [
    {
         "Effect": "Allow",
         "Action": [
```


All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

```
                "signer:*"
            ],
            "Resource": "*"
        }
    ]
}
```

5. Choose **Review policy**.

6. Enter a policy name and description, and then choose **Create policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

7. In the navigation pane, choose **Users**.

8. Choose your IAM user account.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

9. On the **Permissions** tab, choose **Add permissions**.

10. Choose **Attach existing policies directly**, and then select the checkbox next to the code-signing policy you created.

11. Choose **Next: Review**.

12. Choose **Add permissions**.

4 AWS IoT

4.1 Create an AWS IoT thing

1. Open the AWS IoT console website https://console.aws.amazon.com/iot/.
2. On the **Welcome to the AWS IoT Console** page, in the navigation pane, choose **Manage**.

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

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

4. On the **Creating AWS IoT things** page, choose **Create a single thing**.

5. On the **Create a thing** page, in the **Name** field, enter a name for your thing, such as MyThing. Choose **Next**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

6. On the **Add a certificate for your thing** page, choose **Create certificate**. This generates an X.509 certificate and a key pair.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

7. On the **Certificate created!** page, download your public and private keys, certificate, and root certificate authority (CA).
8. Choose **Download** for your certificate.
9. Choose **Download** for your private key.
10. Choose **Download** for the Amazon root CA.
    A new webpage is displayed.
11. Choose **RSA 2048-bit key: Amazon Root CA 1**.
    This opens another webpage with the text of the root CA certificate.
12. Copy this text and paste it into a file named `Amazon_Root_CA_1.pem`.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

Most web browsers save downloaded files into a Downloads directory. You copy these files to a different directory when you run the sample applications. Choose **Activate** to activate the X.509 certificate, and then choose **Attach a policy**.

13. On the **Add a policy for your thing** page, choose **Register Thing**. After you register your thing, create and attach a new policy to the certificate.

4.2 Create an AWS IoT policy

1. In the left navigation pane, choose **Secure**, and then choose **Policies**. On the **You don't have a policy yet** page, choose **Create a policy**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. On the **Create a policy** page, in the **Name** field, enter a name for the policy (for example, MyIotPolicy).
3. In the **Action** field, enter **iot:Connect**.
4. In the **Resource ARN** field, enter *****.
5. Select the **Allow** checkbox.
   This allows all clients to connect to AWS IoT.
6. After you have entered the information for your policy, choose **Create**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

4.3 Attach an AWS IoT policy to a device certificate

1. In the left navigation pane, choose **Secure**, and then choose **Certificates**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

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

3. In **Attach policies to certificate(s)**, select the checkbox next to the policy you created in the previous step, and then choose **Attach**.

4.4 Attach a certificate to a thing

1. In the box for the certificate you created, choose **...** to open a drop-down menu, and then choose **Attach thing**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. In **Attach things to certificate(s)**, select the checkbox next to the thing you registered, and then choose **Attach**.

3. To verify that the thing is attached, select the box for the certificate.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

4. On the **Details** page for the certificate, in the left navigation pane, choose **Things**.

5. To verify that the policy is attached, on the **Details** page for the certificate, in the left navigation pane, choose **Policies**.

5 Configure the device

The related SDK code folder is available here: `SDK_2.x.x_EVK-MIMXRT1060\boards\evkmimxrt1060\aws_examples\ota_demo_enet`.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

5.1 aws_clientcredential.h

1. Open the AWS IoT console website https://console.aws.amazon.com/iot/
2. On the **Welcome to the AWS IoT Console** page, in the navigation pane, choose **Manage – Things** select the previously created **Thing**.

3. In the navigation pane, choose **Interact,** copy the **REST API** endpoint and **IoT Thing** name.
4. Inside the OTA project, open amazon-freertos – demos – include – aws_clientcredential.h and set the **REST_API** and **IoT Thing** name obtained in the previous step.

5.2 aws_clientcredential_keys.h

1. In your PC, go to …\rtos\amazon-freertos\tools\certificate_configuration.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. opens a web browser the CertificateConfigurator.html.

3. Browse to the Certificate and Key files previously downloaded from your Thing and click **Generate and save aws_clientcredential_keys.h**.

4. Using windows explorer replace the…\rtos\amazon-freertos\demos\include \aws_clientcredential_keys.h with the file obtained in the previous step.

5.3 aws_ota_codesigner_certificate

1. Open the ecdsaigner.crt file using a text editor.
2. Copy all the content – Paste the information at the OTA project amazon-freertos – demos – include – aws_ota_codesigner_certificate.h in the signingcredentialSIGNING_CERTIFICATE_PEM Note: be sure to add **“ at** the beginning of a line and **\n”** on every line break.

```
/*
 * PEM-encoded code signer certificate
 *
 * Must include the PEM header and footer:
 * "-----BEGIN CERTIFICATE-----\n"
 * "...base64 data...\n"
 * "-----END CERTIFICATE-----\n";
 */
static const char signingcredentialSIGNING_CERTIFICATE_PEM[] =
"-----BEGIN CERTIFICATE-----\n"
"MIIBYTCCAQegAwIBAgIJAKCX9bIhkilFMAoGCCqGSM49BAMCMCMxITAfBgNVBAMM\n"
"GEFsZWphbmRyYS5HdXptYW5AbnhwLmNvbTAeFw0xOTEwMjMxNjMzNDJaFw0yMDEw\n"
"MjIxNjMzNDJaMCMxITAfBgNVBAMMGEFsZWphbmRyYS5HdXptYW5AbnhwLmNvbTBZ\n"
"MBMGByqGSM49AgEGCCqGSM49AwEHA0IABGUghBD51mF1J3wf4LYsQ2VgOaDpg98G\n"
"dNC38FWGS7owT4NC5848JumrD8SonnnXpu77Pt7ShuW39hC3Vdi7z1GjJDAiMAsG\n"
"A1UdDwQEAwIHgDATBgNVHSUEDDAKBggrBgEFBQcDAzAKBggqhkjOPQQDAgNIADBF\n"
"AiEAr0pNzlaMax4arCPNiW9HYFdQTvUGyZdRLcDrUol/LQoCIH2U2REoZ59V7r6z\n"
"CMLfHA+kWq84IjxDUE2OgV60RVvC\n"
"-----END CERTIFICATE-----\n";
```


All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

5.4 Build

1. Click the make button to start building the application.

2. If the build is successful, Zero errors message is printed in build console.

5.5 Programming mcu-boot into flash

1. Set all SW7 positions to off.

2. Locate J1, then move the jumper to **3-4**.

© NXP B.V. 2022. All rights reserved.

3. Connect the board to PC via J9 USB connector.
4. Reset the board using **SW3**, then make sure that your RT1060-EVK gets enumerated like Human Interface Devices – USB Input device.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

5. Open the Command Prompt window.
6. Execute the following commands. It is recommended, but not required, to have bash interpreter at hand. Git bash does the job https://gitforwindows.org/ > cd ..\OTA_Bootloader_Scripts-4e081f\OTA_Bootloader_Scripts_0.5 > generate_ota_bootloader_and_program_it_to_flash.sh.
7. Disconnect the USB cable from the J9 USB connector.
8. Set SWD7[1:4]:0010.
9. Return J1 jumper to the default setting **5-6**.

10. Connect the RT1060-EVK to the PC using the OpenSDA USB connector J41, mimxrt1060-evk, and SCH rev A2. Use some terminal application to connect the virtual com port to see the console.
11. Reset the EVK using SW3 at this moment that you should be able to see bootloader messages being printed on a terminal.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

5.6 Flashing the OTA Agent application

1. Attach the Ethernet cable with Internet connection and local DHCP server.
2. Click the download and debug button to start flashing the device.

3. When the device is flashed, the debug pointer turns green the main entry point.
4. Click the Go button to start running the program.

Double check that there are MQTT AWS messages on the terminal.

5. Stop the debug session.

6 OTA

6.1 Create new image

1. Open main.c.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

2. Go to line 86, then change any of the APP_VERSION macros to a higher number.

```
#define APP_VERSION_MAJOR 0
#define APP_VERSION_MINOR 9
#define APP_VERSION_BUILD 3//2
```

3. Open **Project > Options**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

4. In the Category section, choose **Output Converter**.
5. Change the name of the binary so it matches the version change then click **OK**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

6. Use the make button to build and generate the binary. Look for the binary inside the …boards\evkmimxrt1060\aws_examples\ota_demo_enet\iar\flexspi_nor_debug folder.

6.2 Uploading the binary to the S3 bucket

1. Use AWS console to open the S3 service https://console.aws.amazon.com/s3.
2. Select the previously created bucket.
3. Click **Upload**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

4. Drag and drop the ota_demo_enet_v0_9_3.bin binary.

5. Click **Upload**.

6.3 Create OTA Job

1. Open the AWS IoT console website https://console.aws.amazon.com/iot/.
2. On the **Welcome to the AWS IoT Console** page, in the navigation pane, choose **Manage – Jobs**.
3. Select **Create**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

4. Under **Create an Amazon FreeRTOS Over-the-Air (OTA) update job**, choose **Create OTA update job**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

5. Under **Select devices to update**, choose **Select**. To update a single device, choose the **Things** tab.

6. Select the checkbox next to the IoT thing associated with your device. Choose **Next**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

7. Under **Select and sign your firmware image**, choose **Sign a new firmware image for me**.

8. Under **Code signing profile**, choose **Create**.

9. In **Create a code signing profile**, enter a name for your code-signing profile.
   a. Under **Device hardware platform**, select Windows Simulator.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

b. Under **Code signing certificate**, choose **Import** and browse for the ecda certificate created with AWS CLI.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

c. Under **Pathname of code signing certificate on device**, type the default path /certificates/authcert.pem then click **Create**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

10. Under **Select your firmware image in S3**, choose **Select**.

11. Under **Pathname of firmware image on device**, type the default path /device/updates.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

12. Under **IAM role for OTA update job**, choose the role created in previous steps.

13. Choose **Next**.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

14. Under **Job type**, choose **Your job will complete after deploying to the selected devices/groups (snapshot)**.

15. Enter an ID for your OTA update job that the application must run, before clicking the **Create** button.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

6.4 Running the application

1. Open main.c.

2. Go to line 86, then change any of the APP_VERSION macros to the original value.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

```
#define APP_VERSION_MAJOR 0
#define APP_VERSION_MINOR 9
#define APP_VERSION_BUILD 2
```

3. Make and Download and Debug.
4. When running the application, wait until the message of the OTA State Ready is shown in the serial terminal.

```
60 23602 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
61 24602 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
62 25602 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
63 26602 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
64 27602 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
```

5. The OTA agent waits for an OTA job. Go back to the Create OTA job window and click **Create**.

6. The process starts, you can see a similar output.

```
55 18633 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
56 19633 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
57 20633 [iot_thread] State: Ready  Received: 1   Queued: 1   Processed: 1   Dropped: 0
59 21291 [OTA Task] [prvParseJSONbyModel] Extracted parameter [ streamname: AFR_OTA-906e2011-a543-460 21300 [OTA Task]
 [prvParseJSONbyModel] Extracted parameter [ filepath: /device/updates ]
61 21308 [OTA Task] [prvParseJSONbyModel] Extracted parameter [ filesize: 260608 ]
62 21315 [OTA Task] [prvParseJSONbyModel] Extracted parameter [ fileid: 0 ]
 [prvParseJSONbyModel] Extracted parameter [ sig-sha256-ecdsa: MEUCIQC2HsafgBckf65 21340 [OTA Task] [prvParseJobDoc] J
ob was accepted. Attempting to start transfer.
66 21342 [OTA Task] [INFO ][MQTT][lu] (MQTT connection 2020b468) SUBSCRIBE operation scheduled.
67 21358 [OTA Task] [INFO ][MQTT][lu] (MQTT connection 2020b468, SUBSCRIBE operation 2020b7b8) Wai68 21445 [OTA Task]
[INFO ][MQTT][lu] (MQTT connection 2020b468, SUBSCRIBE operation 2020b7b8) Wai69 21454 [OTA Task] [prvSubscribeToDataS
```

7. Start file transfer.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

```
77 24265 [OTA Task] [OTA-NXP] WriteBlock 0 : 400
78 24269 [OTA Task] [prvIngestDataBlock] Remaining: 254
79 24308 [OTA Task] [prvIngestDataBlock] Received file block 1, size 1024
```

```
928 42555 [OTA Task] [OTA-NXP] WriteBlock 3dc00 : 400
929 42560 [OTA Task] [prvIngestDataBlock] Remaining: 2
930 42634 [iot_thread] State: Active  Received: 317   Queued: 255   Processed: 255   Dropped: 62
931 43634 [iot_thread] State: Active  Received: 317   Queued: 255   Processed: 255   Dropped: 62
932 44634 [iot_thread] State: Active  Received: 317   Queued: 255   Processed: 255   Dropped: 62
933 45048 [OTA Task] [INFO ][MQTT][lu] (MQTT connection 2020b468) MQTT PUBLISH operation queued.
934 45057 [OTA Task] [prvPublishGetStreamMessage] OK: $aws/things/rt1060_test1/streams/AFR_OTA-906
] [prvIngestDataBlock] Received file block 242, size 1024
936 45256 [OTA Task] [OTA-NXP] WriteBlock 3c800 : 400
937 45261 [OTA Task] [prvIngestDataBlock] Remaining: 1
938 45266 [OTA Task] [prvIngestDataBlock] Received file block 252, size 1024
939 45273 [OTA Task] [OTA-NXP] WriteBlock 3f000 : 400
940 45278 [OTA Task] [prvIngestDataBlock] Received final expected block of file.
```

8. Swap.

```
-----------------------
Swap is in progress...
swap_type:kSwapType_Test
swap_progress: offset=0x00000000, scratch_size=0x00000000, stage=kSwapStage_Done, remaining_size=0x00000000
Image Info:image[0].size=0x0x0003fe00, image[1].size=0x0x0003fe00
```

9. Device gets restarted, then the new application starts running.

```
Running bootloader...
Bootloader version K2.7.0
Initing HID
Initializing PHY...
0 124 [Tmr Svc] Write certificate...
1 266 [Tmr Svc] Device credential provisioning succeeded.
2 1946 [Tmr Svc] Getting IP address from DHCP ...
3 4946 [Tmr Svc] IPv4 Address: 10.42.0.218
4 4946 [Tmr Svc] DHCP OK
5 4949 [iot_thread] [INFO ][INIT][lu] SDK successfully initialized.
6 4949 [iot_thread] [INFO ][DEMO][lu] Successfully initialized the de
[INFO ][MQTT][lu] MQTT library successfully initialized.
8 4949 [iot_thread] OTA demo version 0.9.3
9 4949 [iot_thread] Creating MQTT Client...
```

7 Revision history

This table summarizes revisions to this document.

Table 1. Revision history

Revision number	Date	Substantive changes
0	12/2019	Initial release
1	01 June 2020	Updated for MCUXpresso SDK v2.8.0
2	15 June 2022	Layout updated for MCUXpresso SDK v2.12.0. Added a note in the overview section and made some editorial changes.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

8 Legal information

8.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

8.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

8.3 Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2022. All rights reserved.

Contents

1 Overview 2
2 AWS OTA prerequisites 2
2.1 Create an Amazon S3 bucket and store your update 2
2.2 Create an OTA update service role 3
2.2.1 Create an OTA service role 4
2.2.2 To add OTA update permissions to your OTA service role 8
2.2.3 To add the required IAM permissions to your OTA service role 9
2.2.4 To add the required Amazon S3 permissions to your OTA service role 12
2.3 Create an OTA user policy 14
2.4 Windows prerequisites 16
2.4.1 OpenSSL 16
2.4.2 Install the AWS CLI 17
2.5 Creating a code-signing certificate 19
3 Grant access to code signing for AWS IoT 20
4 AWS IoT 23
4.1 Create an AWS IoT thing 23
4.2 Create an AWS IoT policy 27
4.3 Attach an AWS IoT policy to a device certificate 29
4.4 Attach a certificate to a thing 30
5 Configure the device 32
5.1 aws_clientcredential.h 33
5.2 aws_clientcredential_keys.h 33
5.3 aws_ota_codesigner_certificate 34
5.4 Build 35
5.5 Programming mcu-boot into flash 35
5.6 Flashing the OTA Agent application 38
6 OTA 38
6.1 Create new image 38
6.2 Uploading the binary to the S3 bucket 41
6.3 Create OTA Job 42
6.4 Running the application 51
7 Revision history 53
8 Legal information 54

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

© NXP B.V. 2022. **All rights reserved.**

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 15 June 2022
Document identifier: AWSOTAUG
Document number: