OTA with AWS

Prepare the Demo Files

1. In MTBIDE, navigate to `vendors/cypress/boards/cy8ckit-064s0s2-4343w/aws_demos/config_files/aws_demo_config.h`
2. Find the line `#define CONFIG_MQTT_DEMO_ENABLED` and change it to `#define CONFIG_OTA_UPDATE_DEMO_ENABLED`
3. Open the Makefile, found in the aws_demos root directory in MTBIDE.
4. Find the line `OTA_SUPPORT` and set it to `1`

Add An LED to the Project

1. In a text editor, open <*amazon-freertos*>*demos**ota**aws_iot_ota_update_demo.c*
   - This directory does not get pulled in to MTBIDE
2. At the top of the file, with the other #includes, add:

```
#include "cyhal.h"
#include "cybsp.h"
```

Add a Task to Blink an LED

1. Immediately before the function `_initializeOtaDemo`, add:

```
static void task_blinky(void* param)
{
        /* Toggle every 1000ms */
        const TickType_t xDelay = 1000 / portTICK_PERIOD_MS;
        while(1)
        {
                /* Invert the user led state */
                cyhal_gpio_toggle((cyhal_gpio_t) CYBSP_LED_RGB_RED);
                vTaskDelay(xDelay);
        }
}
```

Initialize the Pin and the Task

1. In the function `_initializeOtaDemo`, immediately before the return statement, add:

```
cyhal_gpio_init((cyhal_gpio_t) CYBSP_LED_RGB_RED,
                        CYHAL_GPIO_DIR_OUTPUT,
                        CYHAL_GPIO_DRIVE_STRONG,
                        CYBSP_LED_STATE_OFF);
xTaskCreate(task_blinky, "Blinky Task", (configMINIMAL_STACK_SIZE * 4), NULL, (tskIDLE_PRIORITY + 2), NULL);
```

Create Code Signing Certificate

1. Open the ModusToolbox Shell
2. Generate the private key with the following command:

```
openssl genpkey -algorithm EC -pkeyopt ec_paramgen_curve:P-256 -pkeyopt
ec_param_enc:named_curve -outform PEM -out AWSKey.pem
```

3. Create a text file named *cert_config.txt* in the same directory as *AWSkey.pem* with the following contents:

```
[req]
Prompt = no
distinguished_name = my_dn
[my_dn]
commonName = <user_name>@<domain>.com
[my_exts]
keyUsage = digitalSignature
extendedKeyUsage = codeSigning
```

› Modify *user_name* and *domain* to match your credentials (email that you use to access AWS)

Generate the Code Signing Certificate

1. Run the following instruction in the shell:

```
openssl req -new -x509 -config cert_config.txt -extensions my_exts
-nodes -days 365 -key AWSKey.pem -out AWSKey.crt
```

NOTE: You can enter '.' in response to the prompt

2. Paste the contents of *AWS_Key.crt* in *aws_ota_codesigner_certificate.h* at */demos/include*. Follow the format explained in the file. This is used to verify the signature generated by AWS and streamed with the image to the kit.

```
static const char signingcredentialSIGNING_CERTIFICATE_PEM[] = "-----BEGIN CERTIFICATE-----\n"
"MIIBNjCB3aADAgECAgkAodDUms9VahMwCgYIKoZIzj0EAwIwDjEMMAoGA1UEAwwD\n"
"Jy4nMB4XDTIwMTEyNTIxNTEzMFoXDTIxMTEyNTIxNTEzMFowDjEMMAoGA1UEAwwD\n"
"Jy4nMFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAE5ZdytZqldFUWmUqHiKrJiy3W\n"
"4e1KNL/d9eBMF+A0CuGFuY4gq6w2i42Gm3M4Y9YbrrI2JB8dFtdA8a1BB0pts6Mk\n"
"MCIwCwYDVR0PBAQDAgeAMBMGA1UdJQQMMAoGCCsGAQUFBwMDMAoGCCqGSM49BAMC\n"
"A0gAMEUCIQCqUkrSSi6+nA+e7B8w9zUiWCoYmyZSBuF4C/u87SqiBAIge0p0T5OF\n"
"Jq0pbSVdB2Z2yOAn90bgUVeXJvCfYWmetdw=\n"
"-----END CERTIFICATE-----\n";
```

Push the Certificate to AWS

1. Register the certificate and private key with AWS Certificate Manager (ACM). An ARN is created in this stage and stored in certarn.json. This ARN is needed when you create an OTA job. In the shell:

```
aws acm import-certificate --certificate fileb://AWSKey.crt --
private-key fileb://AWSKey.pem > certarn.json
```

Run the FreeRTOS demo project

1. In MTBIDE, select the project aws_demos in the workspace.
2. From the Quick Panel, select aws_demos Program (KitProg3). This programs the board and the demo application starts running after the programming is finished.
3. You can view the status of the running application in the serial terminal.
4. You will see the LED on the kit blink red if the program succeeds.

Create AWS S3 Bucket, OTA Role, OTA Policies

1. Follow the steps on OTA Update Prerequisites
   - Specifically, follow these in order:
     - Create an Amazon S3 Bucket
       - As a subdirectory in your S3 bucket, create a folder named *SignedImages*
     - Create an OTA Update Service Role
     - Create an OTA User Policy
     - Grant Access to Code Signing for AWS IoT
     - ~~Prerequisites for OTA Updates Using MQTT~~

Modify the OTA Project

1. In Eclipse IDE for Modus Toolbox, navigate to *demos/include/aws_application_version.h*
2. Increment the define `APP_VERSION_MINOR`
3. Open the file *aws_iot_ota_update_demo.c* and change all instances of `CYBSP_LED_RGB_RED` to `CYBSP_LED_RGB_GREEN`
4. Build the application, but do not program
5. Navigate to the S3 console and click on the S3 bucket you created earlier
6. Click on the *SignedImages* folder to open it
7. Select **Upload** followed by **Add files**
8. Navigate to *<freertos>/build/cy/aws_demos/CY8CKIT-064S0S2_4343W/Debug/cm4.bin* and choose **Open** and **Upload**

Create an OTA Update Job

1. In the navigation pane of the AWS IoT console, choose **Manage**, and then choose **Jobs**.
2. Choose **Create**.
3. Under **Create a FreeRTOS Over-the-Air (OTA) update job**, choose **Create OTA update job**.
4. Under Select devices to update, on the right click **Select**. This will list all the devices you have available in your AWS IoT Console.
5. Select the thing that you created. Then click **Next**.
6. Make sure that the **Sign a new firmware image for me** is selected.
7. Under **Code signing profile**, click **create**.
8. Enter a profile name

Cont’d

1. Under **Device hardware platform**, click select and select **Windows Simulator**
2. Under **Code signing certificate** click **import**
3. Select **certificate** and browse to the AWSKey.crt file you created earlier
4. Select **certificate private key** and browse to the corresponding AWSKey.pem file you created earlier
5. Click **import**
6. Under **Pathname of code signing certificate on device**, enter: /certificates/authcert.pem
7. Click **Create**
8. Under **Select firmware image in S3 or upload it**, click select and then your S3 bucket and then navigate to your cm4.bin file in S3

Cont’d

1. In the **Pathname of firmware image on device**, enter
2. `/device/updates`
3. Under the **IAM role for OTA update job** click **Select** and then select the role you created previously
4. Click **Next**
5. Enter an ID for your Job
6. Click **Create**
7. Click **View** to watch the job

Part of your life. Part of tomorrow.