1,Extract device certificate by executing kmwebnet/NXP-SE050-Test-Get-Cert-idf4

https://github.com/kmwebnet/NXP-SE050-Test-Get-Cert-idf4

App :INFO :PlugAndTrust\_v02.16.00\_20200708

App :INFO :If you want to over-ride the selection, use ENV=EX\_SSS\_BOOT\_SSS\_PORT or pass in command line arguments.

sss :INFO :atr (Len=35)

00 A0 00 00 03 96 04 03 E8 00 FE 02 0B 03 E8 08

01 00 00 00 00 64 00 00 0A 4A 43 4F 50 34 20 41

54 50 4F

cert object ID: f0000101

-----BEGIN CERTIFICATE-----

MIICIDCCAcagAwIBAgIUBABQAUpm2HUHeOYEfpEKlGaAAAAwCgYIKoZIzj0EAwIw

XTEXMBUGA1UECwwOUGx1ZyBhbmQgVHJ1c3QxDDAKBgNVBAoMA05YUDE0MDIGA1UE

…

MDE0QTY2RDg3NTA3NzhFNjA0N0U5MTBBOTQ2NjgwMAoGCCqGSM49BAMCA0gAMEUC

IAC+1nwdDfqWGzqUDhhH3GdlKnW2bO8UNLzL67K/fV39AiEAjnI2M1IGj9/LFwET

Ml9uhzigDkuZzeQQDGzoUidu7b8=

-----END CERTIFICATE-----

-----BEGIN UID-----

BABQAUpm2HUHeOYEfpEKlGaA

-----END UID-----

App :INFO :ex\_sss Finished

2,Copy the contents from "-----BEGIN CERTIFICATE-----" to "-----END CERTIFICATE-----" and save it to file com8trustred.crt

3,Install aws cli v2

You must use a version of the AWS CLI that has AWS-Multiaccount-Registration enabled.

<https://docs.aws.amazon.com/cli/latest/userguide/install-cliv2.html>

>aws --version

aws-cli/2.0.35 Python/3.7.7 Windows/10 botocore/2.0.0dev39

>aws configure --profile <user name>

AWS Access Key ID [None]: <access key id>

AWS Secret Access Key [None]: <secret access key>

Default region name [None]: ap-northeast-1

Default output format [None]: json

4,Register certificate

>aws iot register-certificate-without-ca --certificate-pem file://f:\path\to\com8trusted.crt --status ACTIVE --profile <user name>

{

"certificateArn": "arn:aws:iot:ap-northeast-1:683423990865:cert/d37b57ca7caeeefd29b2556547295bd69bb6dd3b586d2c6c1591b8ddfedf809c",

"certificateId": "d37b57ca7caeeefd29b2556547295bd69bb6dd3b586d2c6c1591b8ddfedf809c"

}



5,create test policy

save following config to file wildCardPolicy

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": [

"iot:Connect",

"iot:Publish",

"iot:Receive",

"iot:Subscribe"

],

"Resource": [

"\*"

]

}

]

}

>aws iot create-policy --policy-name wildCardPolicy --policy-document file://"f:\path\to\wildCardPolicy" --profile <user name>

{

"policyName": "wildCardPolicy",

"policyArn": "arn:aws:iot:ap-northeast-1:683423990865:policy/wildCardPolicy",

"policyDocument": "{\n \"Version\": \"2012-10-17\",\n \"Statement\": [\n {\n \"Effect\": \"Allow\",\n \"Action\": [\n \"iot:Connect\",\n \"iot:Publish\",\n \"iot:Receive\",\n \"iot:Subscribe\"\n ],\n \"Resource\": [\n \"\*\"\n ]\n }\n ]\n}\n",

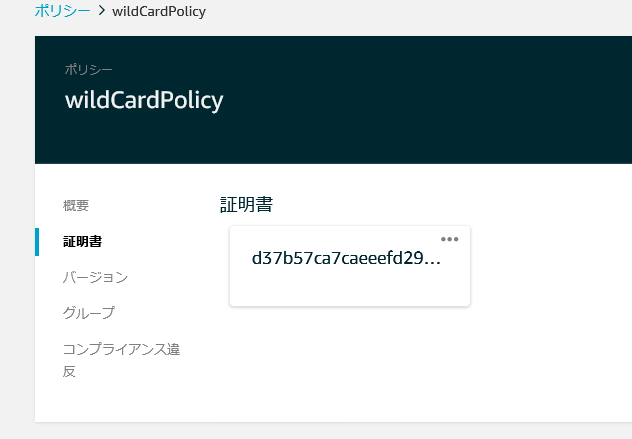
"policyVersionId": "1"

}



6,Attach policy to registered certificate by specifying certificateArn.

>aws iot attach-policy --target "arn:aws:iot:ap-northeast-1:683423990865:cert/d37b57ca7caeeefd29b2556547295bd69bb6dd3b586d2c6c1591b8ddfedf809c" --policy-name wildCardPolicy --profile <user name>



7,Get AWS IoT connection endpoint URL

>aws iot describe-endpoint --endpoint-type iot:Data-ATS --profile <user name>

{

"endpointAddress": "a1xxxxxxxxx-ats.iot.ap-northeast-1.amazonaws.com"

}

8,Replace char HostAddress[255] value on main.c of kmwebnet/NXP-SE050-AWS-IoT-Multi-Account-Registration-Test-idf4 and compile & upload.

https://github.com/kmwebnet/NXP-SE050-AWS-IoT-Multi-Account-Registration-Test-idf4

9,You can get the result on ESP32 console as follows:

App :INFO :If you want to over-ride the selection, use ENV=EX\_SSS\_BOOT\_SSS\_PORT or pass in command line arguments.

sss :INFO :atr (Len=35)

00 A0 00 00 03 96 04 03 E8 00 FE 02 0B 03 E8 08

01 00 00 00 00 64 00 00 0A 4A 43 4F 50 34 20 41

54 50 4F

sss :INFO :Group id found - MBEDTLS\_ECP\_DP\_SECP256R1

I (9991) subpub: Subscribing...

I (10251) subpub: Stack remaining for task 'aws\_iot\_task' is 51392 bytes

I (15311) subpub: Subscribe callback

I (15311) subpub: test\_topic/esp32 hello from ESP32 (QOS0) : 0 : BABQAUpm2HUHeOYEfpEKlGaA

I (15341) subpub: Subscribe callback

I (15341) subpub: test\_topic/esp32 hello from ESP32 (QOS1) : 1 : BABQAUpm2HUHeOYEfpEKlGaA

I (15431) subpub: Stack remaining for task 'aws\_iot\_task' is 51392 bytes

I (20491) subpub: Subscribe callback

I (20491) subpub: test\_topic/esp32 hello from ESP32 (QOS0) : 2 : BABQAUpm2HUHeOYEfpEKlGaA

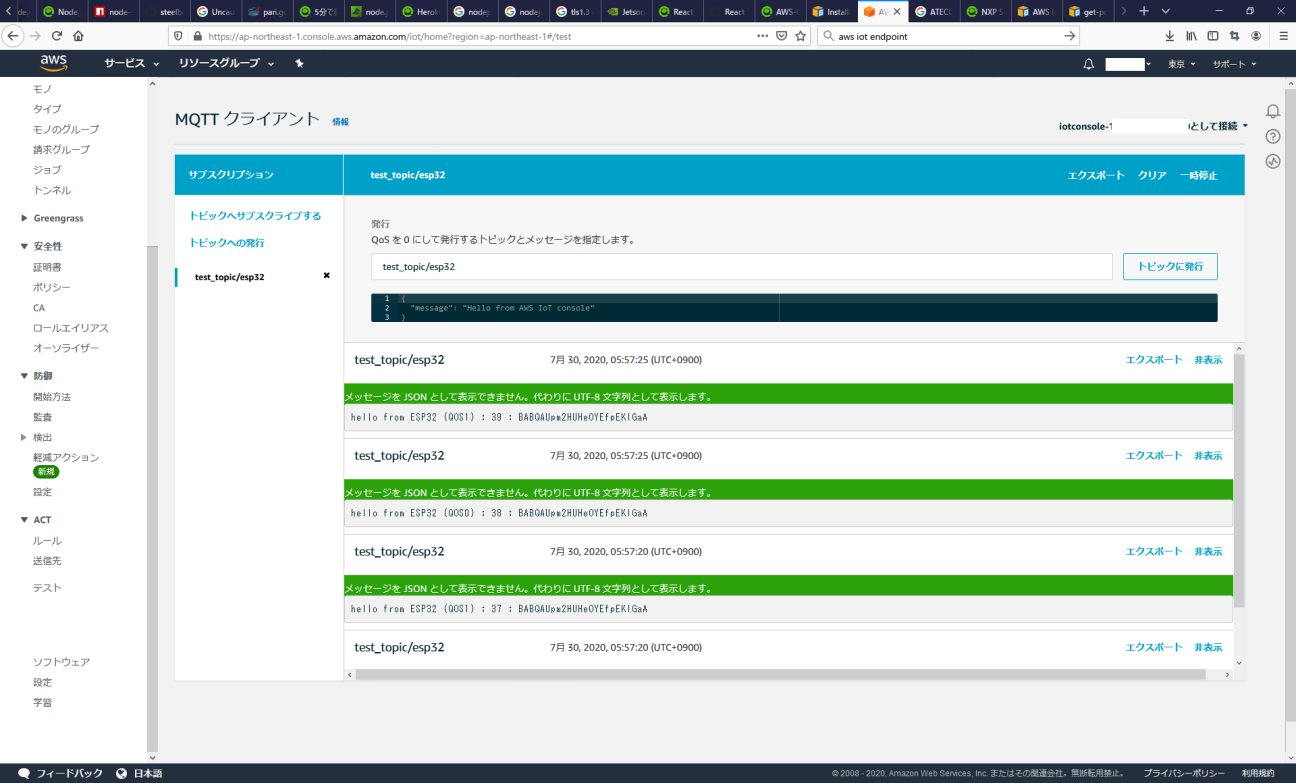
I (20521) subpub: Subscribe callback

I (20521) subpub: test\_topic/esp32 hello from ESP32 (QOS1) : 3 : BABQAUpm2HUHeOYEfpEKlGaA

I (20611) subpub: Stack remaining for task 'aws\_iot\_task' is 51392 bytes

10,You can confirm the connection AWS WEB as follows.

If you press the "Publish to Topic" button and publish from AWS, you can confirm that you are subscribed to the ESP32 console as well.



I (590481) subpub: Subscribe callback

I (590481) subpub: test\_topic/esp32 {

"message": "Hello from AWS IoT console"

}