Asset Tracker v2

Contents

[Reference Variables 3](#_Toc165302110)

[#1 Go to Asset Tracker v2 4](#_Toc165302111)

[#2 Application Description 5](#_Toc165302112)

[#3 Setting up Cloud Service 6](#_Toc165302113)

[#4 Setting up AWS IoT Core 7](#_Toc165302114)

[#5 Step (1) Setting up AWS and configuring permissions 8](#_Toc165302115)

[#6 Step (2) Generating and provisioning certificates 9](#_Toc165302116)

[#6B Step (2) – SubStep (1) – Obtain a list of installed keys using the following command 10](#_Toc165302117)

[#6B Step (2) – SubStep (2) – Obtain a list of installed keys using the following command 11](#_Toc165302118)

[#6B Step (2) – SubStep (3) – Generate a key pair and obtain a CSR using the following command 12](#_Toc165302119)

[#6B Step (2) – SubStep (4) – Convert the CSR from DER format to PEM format using the following command: 13](#_Toc165302120)

[#6B Step (2) – SubStep (5) – Obtain a signed certificate using the following command 14](#_Toc165302121)

[#6B Step (2) – SubStep (6) - Take note of the certificate ARN, as it will be required later 15](#_Toc165302122)

[#6B Step (2) – SubStep (7) - Provision the certificate using the following command: 16](#_Toc165302123)

[#6B Step (2) – SubStep (8) - Download the Amazon Root CA 1 PEM file. 17](#_Toc165302124)

[#6B Step (2) – SubStep (9) - Provision the certificate using the following command: 18](#_Toc165302125)

[#6B Step (3) – SubStep (1) - Create a file policy.json with the following content: 19](#_Toc165302126)

[#6B Step (3) – SubStep (2) - Create the policy using the following command: 21](#_Toc165302127)

[#6B Step (3) – SubStep (3) - Attach the policy to the previously registered certificate using the following command: 22](#_Toc165302128)

[#6B Step (4) – SubStep (1) – Create a Thing – Create a Thing using the following command 23](#_Toc165302129)

[#6B Step (4) – SubStep (2) – Create a Thing – Attach the certificate to the *Thing* using the following command: 24](#_Toc165302130)

[#6B Step (5) – SubStep (1) – Configuring the library – Obtain the AWS IoT broker endpoint using the following command: 25](#_Toc165302131)

[#6B Step (5) – SubStep (2 to 4) – Configuring the library et the CONFIG\_AWS\_IOT\_BROKER\_HOST\_NAME 26](#_Toc165302132)

[#3 Application description – Configuring the Application 28](#_Toc165302133)

[#3 Application description – Configuring the Application – substep (1) Update the overlay…. 29](#_Toc165302134)

[#3 Application description – Configuring the Application – substep (2) Include the overlay.. 30](#_Toc165302135)

[#7 Cellular AT Client 31](#_Toc165302136)

[#8 Cellular AT Client – Testing 32](#_Toc165302137)

[#9 Cellular AT Client – Step (1) Testing Press Reset 33](#_Toc165302138)

[#10 Cellular AT Client – Step (2) Connect to the nRF91 Series with nRF Connect Serial Terminal 35](#_Toc165302139)

[#10 Cellular AT Client – Step (3) Run the following commands from the Serial Terminal 37](#_Toc165302140)

[Installations 38](#_Toc165302141)

[# Install OpenSSL on Windows 38](#_Toc165302142)

[Problem Tickets 39](#_Toc165302143)

[#1 ERROR\_20240428\_1 #10 Cellular AT Client – Step (3) Run the following commands from the Serial Terminal - Case ID: 326042 39](#_Toc165302144)

[#2 ERROR\_20240428\_2 Trying to configure the aws\_iot example for the asset tracker v2 - Case ID: 326050 40](#_Toc165302145)

[#3 ERROR\_20240428\_2 Trying to configure the aws\_iot example for the asset tracker v2 - Case ID: 326122 40](#_Toc165302146)

# Reference Variables

**Client ID : ??? 1051276485**

**IMEI: 355025930003064**

**PIN: 590845**

**aws\_iot project prf.conf**

CONFIG\_MQTT\_HELPER\_SEC\_TAG=201

**Certificate ARN**

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>aws iot create-certificate-from-csr --certificate-signing-request file://device\_cert.csr.pem --certificate-pem-outfile device\_cert.pem --set-as-active --no-cli-pager --query certificateArn

"arn:aws:iot:us-west-1:274488353815:cert/582c3f9cafa73cac9de3217fe074a19835e76799a6486338b6d9739b962d659b"

**Note:** The CONFIG\_AWS\_IOT\_BROKER\_HOST NAME = "a3803pm3fycy7p-ats.iot.us-west-1.amazonaws.com"

**Note:** The CONFIG\_AWS\_IOT\_CLIENT\_ID\_STATIC = "pagps-dev-01",

**Note:** The CONFIG\_MQTT\_HELPER\_SEC\_TAG = 201

# #1 Go to Asset Tracker v2

URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/applications/asset_tracker_v2/README.html>

CLICK:

A screenshot of a computer

Description automatically generated

## #2 Application Description

<https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/applications/asset_tracker_v2/doc/asset_tracker_v2_description.html>

CLICK:

A screenshot of a computer

Description automatically generated

### #3 Setting up Cloud Service

<https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/applications/asset_tracker_v2/doc/asset_tracker_v2_description.html#cloud-setup>

A screenshot of a computer

Description automatically generated

**Client ID : ??? 1051276485**

**IMEI: 355025930003064**

**PIN: 590845**

#### #4 Setting up AWS IoT Core

<https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html#lib-aws-iot>

A screenshot of a computer

Description automatically generated

##### #5 Step (1) Setting up AWS and configuring permissions

A screenshot of a computer

Description automatically generated

This step has been Completed.

##### #6 Step (2) Generating and provisioning certificates

A screenshot of a computer

Description automatically generated

##### #6B Step (2) – SubStep (1) – Obtain a list of installed keys using the following command

SAME URL: https://developer.nordicsemi.com/nRF\_Connect\_SDK/doc/latest/nrf/libraries/networking/aws\_iot.html#lib-aws-iot A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: nrfcredstore <serial port> list

A computer screen shot of a code

Description automatically generated

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>nrfcredstore COM9 list

Secure tag Key type SHA

16842753 ROOT\_CA\_CERT D8D81DE9D4DC85EC920862EB3BFAF1340BEE68E8505336C2E90F2E78ABF58D85

16842753 CLIENT\_CERT 90E2E888EC3F96B282B032164AFF9C37AE228ECD97B868915244B865E5DF3C88

16842753 CLIENT\_KEY 8B0968DFC9742D6648ABFEE3E4E973E2F09D9226C1F817686ED9202A0F791F24

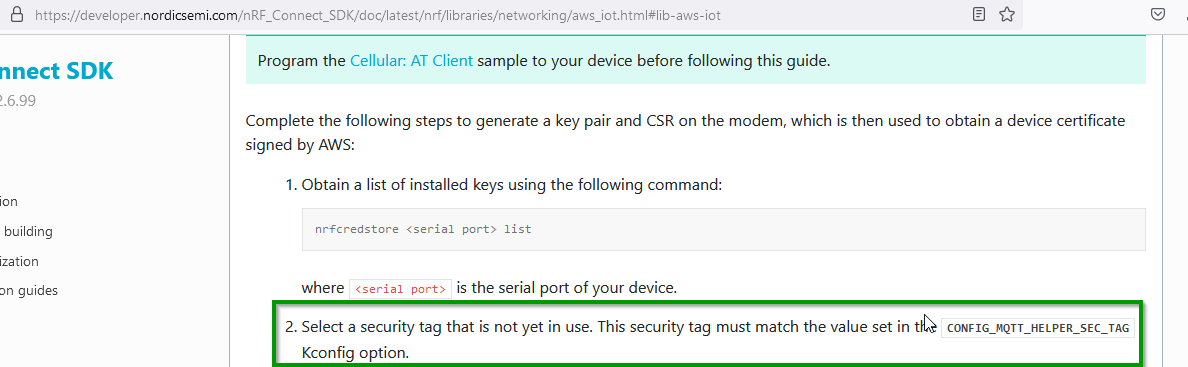
4294967293 NORDIC\_ID\_ROOT\_CA 2C43952EE9E000FF2ACC4E2ED0897C0A72AD5FA72C3D934E81741CBD54F05BD1

4294967294 DEV\_ID\_PUB\_KEY 2B12CF21FE11106722DCD4F11A7BB97B54458C9EE3FC89DAD7B181D21059072B

4294967292 NORDIC\_PUB\_KEY 672E2F05962B4EFBFA8801255D87E0E0418F2DDF4DDAEFC59E9B4162F512CB63

##### #6B Step (2) – SubStep (2) – Obtain a list of installed keys using the following command

SAME URL: https://developer.nordicsemi.com/nRF\_Connect\_SDK/doc/latest/nrf/libraries/networking/aws\_iot.html



**NOTE:** The CONFIG\_MQTT\_HELPER\_SEC\_TAG is located in the in the aws\_iot project in the prj.conf file

A screenshot of a computer program

Description automatically generated

CONFIG\_MQTT\_HELPER\_SEC\_TAG=201

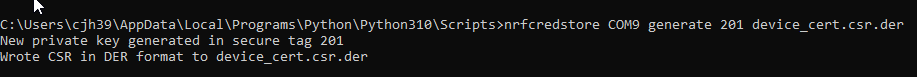
##### #6B Step (2) – SubStep (3) – Generate a key pair and obtain a CSR using the following command

SAME URL: https://developer.nordicsemi.com/nRF\_Connect\_SDK/doc/latest/nrf/libraries/networking/aws\_iot.html

A screenshot of a computer

Description automatically generatedDIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: nrfcredstore <serial port> generate <sec tag> device\_cert.csr.der



**NOTE:** The 201 comes from the last instruction , the CONFIG\_MQTT\_HELPER\_SEC\_TAG=201

The file device\_cert.csr.der is located in C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

##### #6B Step (2) – SubStep (4) – Convert the CSR from DER format to PEM format using the following command:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: openssl req -inform DER -in device\_cert.csr.der -outform PEM -out device\_cert.csr.pem

A computer screen with white text

Description automatically generated

C:\Program Files\OpenSSL-Win64\bin>openssl.exe req -inform DER -in \Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts\device\_cert.csr.der -outform PEM -out \Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts\device\_cert.csr.pem

##### #6B Step (2) – SubStep (5) – Obtain a signed certificate using the following command

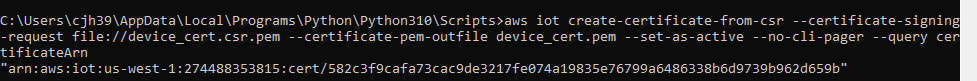
SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: aws iot create-certificate-from-csr --certificate-signing-request file://device\_cert.csr.pem --certificate-pem-outfile device\_cert.pem --set-as-active --no-cli-pager --query certificateArn



C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>aws iot create-certificate-from-csr --certificate-signing-request file://device\_cert.csr.pem --certificate-pem-outfile device\_cert.pem --set-as-active --no-cli-pager --query certificateArn

"arn:aws:iot:us-west-1:274488353815:cert/582c3f9cafa73cac9de3217fe074a19835e76799a6486338b6d9739b962d659b"

##### #6B Step (2) – SubStep (6) - Take note of the certificate ARN, as it will be required later

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

"arn:aws:iot:us-west-1:274488353815:cert/582c3f9cafa73cac9de3217fe074a19835e76799a6486338b6d9739b962d659b"

##### #6B Step (2) – SubStep (7) - Provision the certificate using the following command:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: nrfcredstore <serial port> write <sec tag> CLIENT\_CERT device\_cert.pem

A screen shot of a computer program

Description automatically generated

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>nrfcredstore COM9 write 201 CLIENT\_CERT device\_cert.pem

**NOTE:** The 201 comes from the last instruction , the CONFIG\_MQTT\_HELPER\_SEC\_TAG=201

##### #6B Step (2) – SubStep (8) - Download the [Amazon Root CA 1](https://www.amazontrust.com/repository/AmazonRootCA1.pem) PEM file.

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

URL: <https://www.amazontrust.com/repository/AmazonRootCA1.pem>

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

MIIDQTCCAimgAwIBAgITBmyfz5m/jAo54vB4ikPmljZbyjANBgkqhkiG9w0BAQsF

ADA5MQswCQYDVQQGEwJVUzEPMA0GA1UEChMGQW1hem9uMRkwFwYDVQQDExBBbWF6

b24gUm9vdCBDQSAxMB4XDTE1MDUyNjAwMDAwMFoXDTM4MDExNzAwMDAwMFowOTEL

MAkGA1UEBhMCVVMxDzANBgNVBAoTBkFtYXpvbjEZMBcGA1UEAxMQQW1hem9uIFJv

b3QgQ0EgMTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALJ4gHHKeNXj

ca9HgFB0fW7Y14h29Jlo91ghYPl0hAEvrAIthtOgQ3pOsqTQNroBvo3bSMgHFzZM

9O6II8c+6zf1tRn4SWiw3te5djgdYZ6k/oI2peVKVuRF4fn9tBb6dNqcmzU5L/qw

IFAGbHrQgLKm+a/sRxmPUDgH3KKHOVj4utWp+UhnMJbulHheb4mjUcAwhmahRWa6

VOujw5H5SNz/0egwLX0tdHA114gk957EWW67c4cX8jJGKLhD+rcdqsq08p8kDi1L

93FcXmn/6pUCyziKrlA4b9v7LWIbxcceVOF34GfID5yHI9Y/QCB/IIDEgEw+OyQm

jgSubJrIqg0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8BAf8EBAMC

AYYwHQYDVR0OBBYEFIQYzIU07LwMlJQuCFmcx7IQTgoIMA0GCSqGSIb3DQEBCwUA

A4IBAQCY8jdaQZChGsV2USggNiMOruYou6r4lK5IpDB/G/wkjUu0yKGX9rbxenDI

U5PMCCjjmCXPI6T53iHTfIUJrU6adTrCC2qJeHZERxhlbI1Bjjt/msv0tadQ1wUs

N+gDS63pYaACbvXy8MWy7Vu33PqUXHeeE6V/Uq2V8viTO96LXFvKWlJbYK8U90vv

o/ufQJVtMVT8QtPHRh8jrdkPSHCa2XV4cdFyQzR1bldZwgJcJmApzyMZFo6IQ6XU

5MsI+yMRQ+hDKXJioaldXgjUkK642M4UwtBV8ob2xJNDd2ZhwLnoQdeXeGADbkpy

rqXRfboQnoZsG4q5WTP468SQvvG5

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

Downloaded to C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

##### #6B Step (2) – SubStep (9) - Provision the certificate using the following command:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: nrfcredstore <serial port> write <sec tag> ROOT\_CA\_CERT AmazonRootCA1.pem

A screen shot of a computer program

Description automatically generated

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts> nrfcredstore COM9 write 201 ROOT\_CA\_CERT AmazonRootCA1.pem

**NOTE:** The 201 comes from the last instruction , the CONFIG\_MQTT\_HELPER\_SEC\_TAG=201

##### #6B Step (3) – SubStep (1) - Create a file policy.json with the following content:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

We are using the following policy:

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>type policy.json

{

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

"Statement": [

{

"Effect": "Allow",

"Action": "iot:Connect",

"Resource": "arn:aws:iot:us-east-2:274488353815:client/thinggps"

},

{

"Effect": "Allow",

"Action": "iot:Publish",

"Resource": [

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/update",

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/delete",

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/get"

]

},

{

"Effect": "Allow",

"Action": "iot:Receive",

"Resource": [

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/update/accepted",

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/delete/accepted",

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/get/accepted",

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/update/rejected",

"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/delete/rejected"

]

},

{

"Effect": "Allow",

"Action": "iot:Subscribe",

"Resource": [

"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/update/accepted",

"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/delete/accepted",

"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/get/accepted",

"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/update/rejected",

"arn:aws:iot:us-west-1:274488353815:topicfilter/$aws/thing/thinggps/shadow/delete/rejected"

]

},

{

"Effect": "Allow",

"Action": [

"iot:GetThingShadow",

"iot:UpdateThingShadow",

"iot:DeleteThingShadow"

],

"Resource": "arn:aws:iot:us-east-2:274488353815:thing/thinggps"

}

]

}

##### #6B Step (3) – SubStep (2) - Create the policy using the following command:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: aws iot create-policy --policy-name my-policy --policy-document file://policy.json

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>aws iot create-policy --policy-name my-policy --policy-document file://policy.json

{

"policyName": "my-policy",

"policyArn": "arn:aws:iot:us-west-1:274488353815:policy/my-policy",

"policyDocument": "{\n \"Version\": \"2012-10-17\",\n \"Statement\": [\n {\n \"Effect\": \"Allow\",\n \"Action\": \"iot:Connect\",\n \"Resource\": \"arn:aws:iot:us-east-2:274488353815:client/thinggps\"\n },\n {\n \"Effect\": \"Allow\",\n \"Action\": \"iot:Publish\",\n \"Resource\": [\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/update\",\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/delete\",\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/get\"\n ]\n },\n {\n \"Effect\": \"Allow\",\n \"Action\": \"iot:Receive\",\n \"Resource\": [\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/update/accepted\",\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/delete/accepted\",\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/get/accepted\",\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/update/rejected\",\n \"arn:aws:iot:us-east-2:274488353815:topic/$aws/thing/thinggps/shadow/delete/rejected\"\n ]\n },\n {\n \"Effect\": \"Allow\",\n \"Action\": \"iot:Subscribe\",\n \"Resource\": [\n \"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/update/accepted\",\n \"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/delete/accepted\",\n \"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/get/accepted\",\n \"arn:aws:iot:us-east-2:274488353815:topicfilter/$aws/thing/thinggps/shadow/update/rejected\",\n \"arn:aws:iot:us-west-1:274488353815:topicfilter/$aws/thing/thinggps/shadow/delete/rejected\"\n ]\n },\n {\n \"Effect\": \"Allow\",\n \"Action\": [\n \"iot:GetThingShadow\",\n \"iot:UpdateThingShadow\",\n \"iot:DeleteThingShadow\"\n ],\n \"Resource\": \"arn:aws:iot:us-east-2:274488353815:thing/thinggps\"\n }\n ]\n}",

"policyVersionId": "1"

}

##### #6B Step (3) – SubStep (3) - Attach the policy to the previously registered certificate using the following command:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

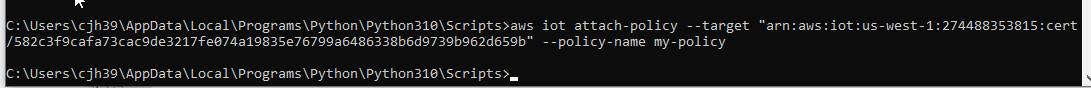
A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: aws iot attach-policy --target <certificate arn> --policy-name my-policy

"arn:aws:iot:us-west-1:274488353815:cert/582c3f9cafa73cac9de3217fe074a19835e76799a6486338b6d9739b962d659b"



C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>aws iot attach-policy --target "arn:aws:iot:us-west-1:274488353815:cert/582c3f9cafa73cac9de3217fe074a19835e76799a6486338b6d9739b962d659b" --policy-name my-policy

##### #6B Step (4) – SubStep (1) – Create a Thing – Create a Thing using the following command

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: aws iot create-thing --thing-name <thing name>

**Note:** We will make the <thing name> pagps-dev-01

A computer screen with text on it

Description automatically generated

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>aws iot create-thing --thing-name pagps-dev-01

{

"thingName": "pagps-dev-01",

"thingArn": "arn:aws:iot:us-west-1:274488353815:thing/pagps-dev-01",

"thingId": "22592e85-6650-4e9c-81b0-e6367c2068bd"

}

##### #6B Step (4) – SubStep (2) – Create a Thing – Attach the certificate to the Thing using the following command:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

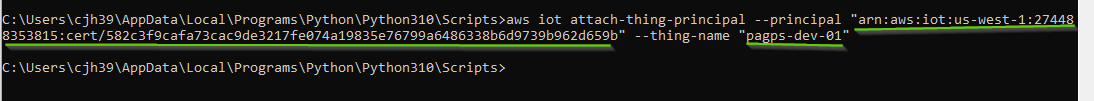
EXECUTE: aws iot attach-thing-principal --principal <certificate arn> --thing-name <thing name>

**Note:** <certificate arn> is **"arn:aws:iot:us-west-1:274488353815:cert/582c3f9cafa73cac9de3217fe074a19835e76799a6486338b6d9739b962d659b"**

<thing name> is "pagps-dev-01"

A screen shot of a computer

Description automatically generated



C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>aws iot attach-thing-principal --principal "arn:aws:iot:us-west-1:274488353815:cert/582c3f9cafa73cac9de3217fe074a19835e76799a6486338b6d9739b962d659b" --thing-name "pagps-dev-01"

##### #6B Step (5) – SubStep (1) – Configuring the library – Obtain the AWS IoT broker endpoint using the following command:

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

DIR: C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts

EXECUTE: aws iot describe-endpoint --endpoint-type iot:Data-ATS

A computer screen with text on it

Description automatically generated

C:\Users\cjh39\AppData\Local\Programs\Python\Python310\Scripts>aws iot describe-endpoint --endpoint-type iot:Data-ATS

{

"endpointAddress": "a3803pm3fycy7p-ats.iot.us-west-1.amazonaws.com"

}

##### #6B Step (5) – SubStep (2 to 4) – Configuring the library et the [CONFIG\_AWS\_IOT\_BROKER\_HOST\_NAME](https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/kconfig/index.html#CONFIG_AWS_IOT_BROKER_HOST_NAME)

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/libraries/networking/aws_iot.html>

A screenshot of a computer

Description automatically generated

**Note:** The CONFIG\_AWS\_IOT\_BROKER\_HOST NAME = "a3803pm3fycy7p-ats.iot.us-west-1.amazonaws.com"

**Note:** The CONFIG\_AWS\_IOT\_CLIENT\_ID\_STATIC = "pagps-dev-01",

**Note:** The CONFIG\_MQTT\_HELPER\_SEC\_TAG = 201

Set these values in the AWS\_IOT projectA screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

### #3 Application description – Configuring the Application

https://developer.nordicsemi.com/nRF\_Connect\_SDK/doc/latest/nrf/applications/asset\_tracker\_v2/doc/asset\_tracker\_v2\_description.html#atv2-application-configuration

A screenshot of a computer

Description automatically generated

#### #3 Application description – Configuring the Application – substep (1) Update the overlay….

URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/applications/asset_tracker_v2/doc/asset_tracker_v2_description.html#atv2-application-configuration>

A screenshot of a computer

Description automatically generated

**Note:** The cloud service selected AWS IoT

**Note:** The CONFIG\_AWS\_IOT\_BROKER\_HOST NAME = "a3803pm3fycy7p-ats.iot.us-west-1.amazonaws.com"

**Note:** The CONFIG\_AWS\_IOT\_CLIENT\_ID\_STATIC = "pagps-dev-01",

**Note:** The CONFIG\_MQTT\_HELPER\_SEC\_TAG = 201

The overlay file is overlay-aws.conf located here:

A screenshot of a computer program

Description automatically generated

The CONFIG\_MQTT\_HELPER\_SEC\_TAG is set to 102

The CONFI\_AWS\_IOT\_BROKER\_HOST\_NAME = "a3803pm3fycy7p-ats.iot.us-west-1.amazonaws.com"

#### #3 Application description – Configuring the Application – substep (2) Include the overlay..

URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/applications/asset_tracker_v2/doc/asset_tracker_v2_description.html#atv2-application-configuration>

A screenshot of a computer

Description automatically generated

The overlay-aws.conf was added to the build configuration Kconfig fragments

A screenshot of a computer

Description automatically generated

## #7 Cellular AT Client

<https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/samples/cellular/at_client/README.html#at-client-sample>

source : samples/cellular/at\_client

A computer screen shot of a black screen

Description automatically generated

#8 Cellular AT Client – Testing

SAME URL: <https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/samples/cellular/at_client/README.html#at-client-sample>

A screenshot of a computer

Description automatically generated

#9 Cellular AT Client – Step (1) Testing Press Reset

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

#10 Cellular AT Client – Step (2) Connect to the nRF91 Series with nRF Connect Serial Terminal

SAME URL: https://developer.nordicsemi.com/nRF\_Connect\_SDK/doc/latest/nrf/samples/cellular/at\_client/README.html#at-client-sample

A screenshot of a computer

Description automatically generated

Use TeraTerm:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

#10 Cellular AT Client – Step (3) Run the following commands from the Serial Terminal

SAME URL: https://developer.nordicsemi.com/nRF\_Connect\_SDK/doc/latest/nrf/samples/cellular/at\_client/README.html#at-client-sample

ERROR SECTION #ERROR\_20240428\_1

A screenshot of a computer error

Description automatically generated

RETEST SECTION

A screenshot of a computer

Description automatically generated

AT%CMNG=1

%CMNG: 201,0,"D8D81DE9D4DC85EC920862EB3BFAF1340BEE68E8505336C2E90F2E78ABF58D85"

%CMNG: 201,1,"5E52AE7F71959893CC5D11A551108115504BC75779A4CEBC7E88A30851C16571"

%CMNG: 201,2,"F9BCA5DDB65292BEDA8479BBC0E46407D766E4DC5149D6B95A6219F534E0FD8F"

%CMNG: 16842753,0,"D8D81DE9D4DC85EC920862EB3BFAF1340BEE68E8505336C2E90F2E78ABF58D85"

%CMNG: 16842753,1,"90E2E888EC3F96B282B032164AFF9C37AE228ECD97B868915244B865E5DF3C88"

%CMNG: 16842753,2,"8B0968DFC9742D6648ABFEE3E4E973E2F09D9226C1F817686ED9202A0F791F24"

%CMNG: 4294967293,10,"2C43952EE9E000FF2ACC4E2ED0897C0A72AD5FA72C3D934E81741CBD54F05BD1"

%CMNG: 4294967294,6,"2B12CF21FE11106722DCD4F11A7BB97B54458C9EE3FC89DAD7B181D21059072B"

%CMNG: 4294967292,11,"672E2F05962B4EFBFA8801255D87E0E0418F2DDF4DDAEFC59E9B4162F512CB63"

## Installations

### # Install OpenSSL on Windows

URL: https://slproweb.com/products/Win32OpenSSL.html

## Problem Tickets

### #1 ERROR\_20240428\_1 #10 Cellular AT Client – Step (3) Run the following commands from the Serial Terminal - Case ID: 326042

TICKET URL:

https://devzone.nordicsemi.com/f/nordic-q-a/110679/cellular-at-client---step-3-run-the-following-commands-from-the-serial-terminal

A screenshot of a computer error

Description automatically generated

### #2 ERROR\_20240428\_2 Trying to configure the aws\_iot example for the asset tracker v2 - Case ID: 326050

### #3 ERROR\_20240428\_2 Trying to configure the aws\_iot example for the asset tracker v2 - Case ID: 326122