Configuration of Basic Station for Kona Micro Lite Gateway on AWS IoT Core

Introduction

- Reference guide to configure Basic Station for Kona Micro Lite Gateway on AWS IoT Core

- High-level process involves below stages.

1. Setup AWS account and permissions
2. Commissioning of Gateway on AWS IoT Core
3. Uploading Basic Station file
4. Uploading configuration files
5. Uploading security certificates

Setup AWS account & Permissions

- Add an IAM Role for CUPS server

- Add an IAM role that will allow the **Configuration and Update Server (CUPS)** to handle the wireless gateway credentials.

Note:

This procedure needs to be done only once, but must be performed before a LoRaWAN gateway tries to connect with AWS IoT Core for LoRaWAN.

- Go to the **IAM Roles** page on the **IAM console**

- Choose **Create role**.

- On the Create Role page, choose **Another AWS account**.

- For Account ID, enter your **account id**.

Setup AWS account & Permissions(cont.)

- Choose Next: **Permissions**

- In the search box next to Filter policies, enter **AWSIoTWirelessGatewayCertManager**.
  - If the search results show the policy named **AWSIoTWirelessGatewayCertManager**, select it by clicking on the checkbox.

- If the policy does not exist, please create it as follows:
  - Go to the **IAM console**.
  - Choose **Policies** from the navigation pane.
  - Choose **Create Policy**. Then choose the JSON tab to open the policy editor. Replace the existing template with this trust policy document:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "IoTWirelessGatewayCertManager",
            "Effect": "Allow",
            "Action": [
                "iot:CreateKeysAndCertificate",
                "iot:DescribeCertificate",
                "iot:ListCertificates",
                "iot:RegisterCertificate"
            ],
            "Resource": "*"
        }
    ]
}
```

Setup AWS account & Permissions(cont.)

- Once the existing template has been replaced with the trust policy, choose **Review Policy** to open the Review page.

- For Name, enter **AWSIoTWirelessGatewayCertManager**.

Note:

You must not use a different name. This is for consistency with future releases.

- For **Description**, enter a description of your choice

- Choose **Create policy**. You will see a confirmation message showing the policy has been created.

Setup AWS account & Permissions(cont.)

- Choose Next: **Tags**, and then choose Next: **Review**

- In Role name, enter **IoTWirelessGatewayCertManagerRole**, and then choose **Create role**.

Note:

You must not use a different name. This is for consistency with future releases.

- In the confirmation message, choose **IoTWirelessGatewayCertManagerRole** to edit the new role

- In the Summary, choose the **Trust relationships** tab, and then choose **Edit trust relationship**.

Setup AWS account & Permissions(cont.)

- In the Policy Document, change the **Principal property** to represent the IoT Wireless service:

```
"Principal": {
    "Service": "iotwireless.amazonaws.com"
},
```

After you change the Principal property, the complete policy document should look like this:

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

       "Principal": {
         "Service": "iotwireless.amazonaws.com"
       },
       "Action": "sts:AssumeRole",
       "Condition": {}
    }
  ]
}
```

- Choose **Update Trust Policy** to save your changes and exit.
  - At this point, you've created the **IoTWirelessGatewayCertManagerRole** and you won't need to do this again

Setup AWS Account & Permissions(cont.)

- Add IAM role for Destination to AWS IoT Core for LoRaWAN
  - Go to the **IAM console**
  - Choose **Policies** from the navigation pane.
  - Choose **Create Policy**. Then choose the JSON tab to open the policy editor. Replace the existing template with this trust policy document:

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action":
          [
            "iot:DescribeEndpoint",
            "iot:Publish"
          ],
        "Resource": "*"
      }
    ]
}
```

  - Choose **Review Policy** to open the Review page.
    - For Name, enter a name of your choice.
    - For Description, enter a description of your choice.
  - Choose **Create policy**.

Setup AWS Account & Permissions(cont.)

- Now, create a role that will use the policy created in previous slide outlining Setup AWS Account & Permissions .
    - In the IAM console, choose **Roles** from the navigation pane to open the Roles page.
    - Choose **Create Role**.
    - In Select type of trusted entity, choose **Another AWS account**.
    - In Account ID, enter your AWS account ID.
    - Choose Next: **Permissions**
    - Search for your **IAM policy** created in the step above. Type in the policy name to find your policy. Select it.
    - Choose Next: **Tags**
    - Choose Next: Review to open the Review page.
        - For Role name, enter an appropriate name of your choice.
        - For Description, enter a description of your choice.
    - Choose **Create Role**.

Setup AWS Account & Permissions(cont.)

- Update your policy's trust relationship
  - In the IAM console, choose **Roles** from the navigation pane to open the Roles page
  - Enter the name of the role you created earlier in the search window, and click on the role name in the search results
  - Choose the **Trust relationships** tab to navigate to the Trust relationships page.
  - Choose **Edit trust relationship**. The principal AWS role in your trust policy document defaults to root. Replace the existing policy with this:

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "",
        "Effect": "Allow",
        "Principal": {
            "Service": "iotwireless.amazonaws.com"
        },
        "Action": "sts:AssumeRole",
        "Condition": {}
      }
    ]
}
```

  - Choose **Update Trust Policy**

Commissioning of Gateway on AWS IoT Core

1. Login to AWS IoT Core Network Server.
2. Drop down the Services Menu and select **IoT Core**
3. Open up the **Wireless Connectivity** tab and Select **Gateways**
4. Click the **Add Gateway** button
5. Enter the **Gateway EUI,** and **Gateway Name**.
6. Select the correct frequency plan based on your Gateway.
7. Select **Add Gateway**.

Certificates and Keys

- To connect a LoRaWAN gateway to AWS IoT Core, below list of certificates and keys will be required:

1. **cups.crt**
2. **cups.trust.**
3. **cups.key**
4. **cups.uri**
5. **tc.crt**
6. **tc.uri**
7. **tc.trust**
8. **tc.key**

- The next page(s) will cover how to obtain these certificate and key files from AWS, and how to upload them to your gateway.

Gateway Certificates and Keys

- Once the Certificate created and associated with your gateway message is shown, select **Download certificates** to download the certificate (xxxxx.cert.pem) and private key (xxxxxx.private.key). We recommend that you store all the downloaded files in the same folder.
    - Then rename xxxx.cert.pem file to cups.crt and xxxx.private.key to cups.key
    - Create a copy of cups.key and name it tc.key
    - Create a copy of cups.crt and name it tc.crt
- In the section Provisioning credentials, choose **Download server trust certificates** to download the CUPS (cups.trust) and LNS (lns.trust) server trust certificates.
    - Keep the **cups.trust** file as it is
    - Rename the **lns.trust** file to **tc.trust**
- Copy the CUPS and LNS endpoints and save them for use while configuring the gateway.
    - Create **cups.uri** file with CUPS Endpoint URL**: e.g: https://EXAMPLE.cups.lorawan.REGION.amazonaws.com:443**
    - Create **tc.uri** file with LNS Endpoint URL**: e.g: wss://EXAMPLE.gateway.lorawan.REGION.amazonaws.com:443**

Gateway Certificates and Keys

- Make sure that you have the following 8 files from the steps above as you'll need them to configure your gateway:

    - tc.uri
    - tc.trust
    - tc.key
    - tc.crt
    - cups.uri
    - cups.trust
    - cups.key
    - cups.crt

Uploading Basic Station File

- Download latest version of Basic Station binary file for Kona Micro Lite gateway using this link.

- Use **TFTP client** to upload downloaded binary file of Basic Station on Kona Micro Lite gateway as shown in Figure-1 and reboot the gateway.

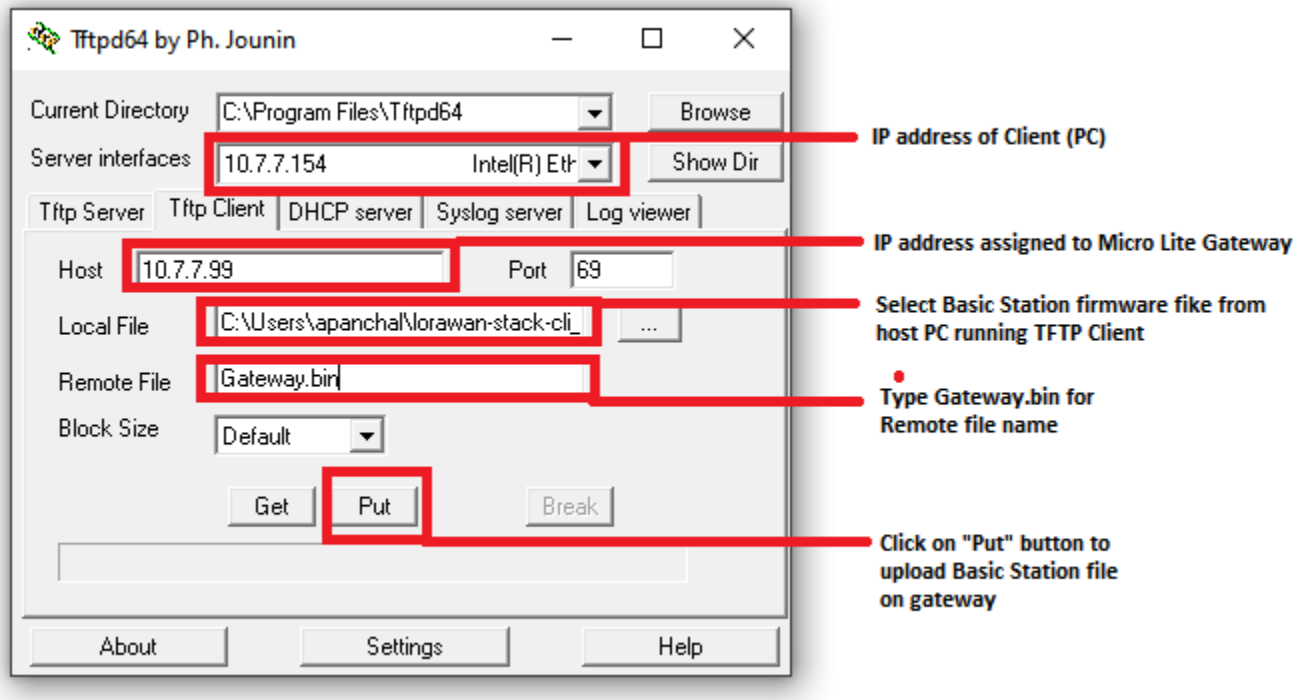

Figure-1 Uploading Basic Station binary

Updating Customer.json file

- To connect Kona Micro Lite Gateway to AWS , you will be required to modify existing **Customer.json** file.

- Modified **Customer.json** file will have a structure as shown below.

```
{
    "private_key_password":"",
    "network":"bstn",
    "bstn":{
        "cups_uri":"<cups_uri_content>",
        "cups_use_token":false,
        "lns_uri":"<lns_uri_content>",
        "lns_use_token":false}
}
```

Figure-3 Customer.json file

- Sample Customer.json file can be downloaded using this link.

- After downloading this file, edit **cups_uri** value by contents of **cups.uri** file.

Uploading Customer.json File

- Use **TFTP client** to upload downloaded binary file of Basic Station on Kona Micro Lite gateway as shown in Figure-2 and reboot the gateway.

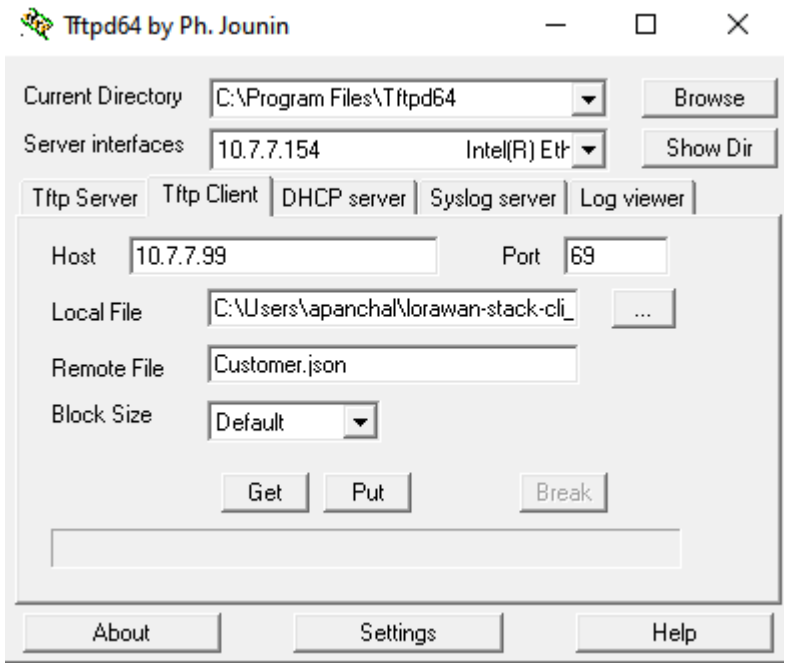

Figure-2 Uploading Customer.json file

CaRootCertificate.pem file

- **CaRootCertificate.pem** file can be generated by copying contents of **cups.trust** and **tc.trust** files.
- To do so, copy contents from **cups.trust** and **tc.trust** files and paste them in a new file.
- Save this new file as **CaRootCertificate.pem** .
- After that, Use TFTP client to upload **CaRootCertificate.pem** file as shown in Figure-3 and reboot the gateway.

CaRootCertificate.pem file

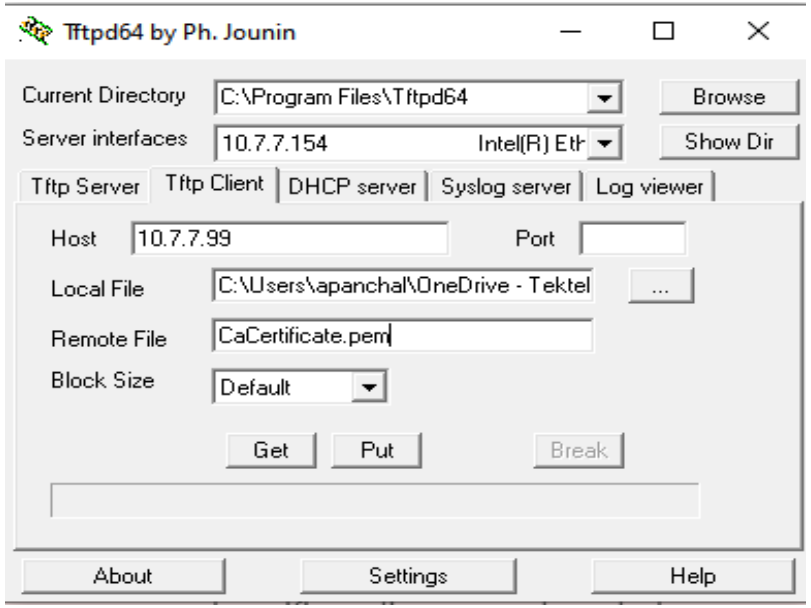

Figure-3 Uploading CaCertificate.pem file

Certificate.pem file

- **Certificate.pem** file can be generated by copying contents of **cups.crt** and **tc.crt** files.
- To do so, copy contents from **cups.crt** and **tc.crt** files and paste them in a new file.
- Save this new file as **Certificate.pem** .
- After that, Use TFTP client to upload **Certificate.pem** file as shown in Figure-4 and reboot the gateway.

CaRootCertificate.pem file

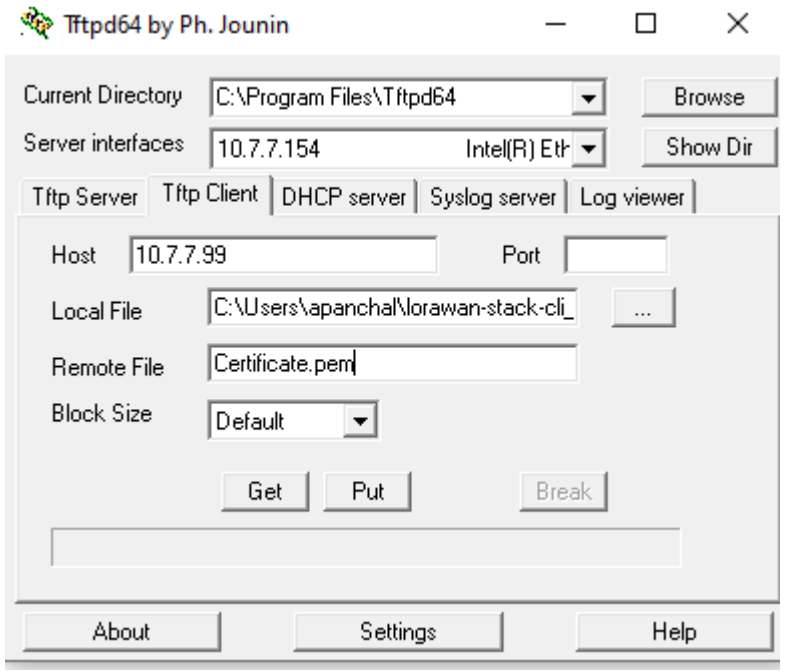

Figure-4 Uploading Certificate.pem file

PrivateKey.pem File

- **PrivateKey.pem** file can be generated by copying contents of **cups.key** and **tc.key** files.
- To do so, copy contents from **cups.key** and **tc.key** files and paste them in a new file.
- Save this file as **PrivateKey.pem** .
- After that, Use TFTP client to upload **PrivateKey.pem** file on Kona Micro Lite gateway as shown in Figure-5 and reboot the gateway.

PrivateKey.pem File (Cont…)

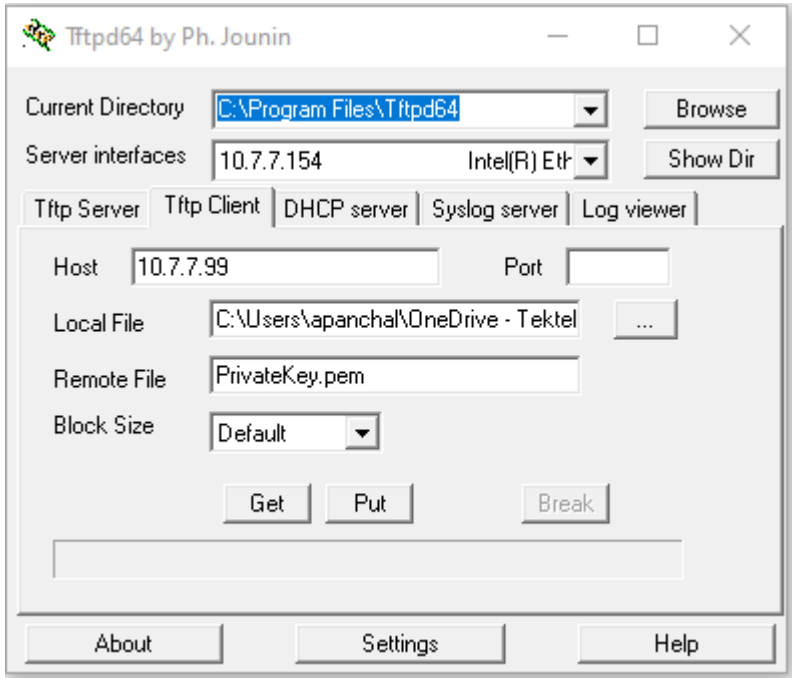

Figure-5 Uploading PrivateKey.pem file

- After reboot, you will be able to see gateway showing Connected on AWS.

Best-In-Class, Carrier Grade & Most Cost Effective Portfolio of Gateways, Network Server, Sensors & Applications