Application Note AZURE_prov_X.509 Example

Version 1.0.0

For more information, visit our website at http://www.wiznet.io

Contents

Figures

Tables

1 Introduction

This Application Note covers the implementation of AZURE prov X.509 authentication and the generation of X.509 certificates on WIZnet's TOE Chip.

2 Github Link

https://github.com/WIZnet-ioNIC/WIZnet-PICO-AZURE-C.git

3 Applicable products

Raspberry Pi Pico & WIZnet Ethernet HAT
W5100S-EVB-Pico
W5500-EVB-Pico
W55RP20-EVB-Pico
W5100S-EVB-Pico2
W5500-EVB-Pico2

4 How to Test AZURE prov X.509 Example

4.1 Step 1: Prepare software

The following serial terminal program is required for AZURE prov X.509 example test, download and install from below links.

- **Tera Term**

4.2 Step 2: Prepare hardware

If you are using W5100S-EVB-Pico, W5500-EVB-Pico, W55RP20-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2, you can skip '1. Combine...'

1. Combine WIZnet Ethernet HAT with Raspberry Pi Pico.
2. Connect ethernet cable to WIZnet Ethernet HAT, W5100S-EVB-Pico, W5500-EVB-Pico, W55RP20-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2 ethernet port.
3. Connect Raspberry Pi Pico, W5100S-EVB-Pico or W5500-EVB-Pico to desktop or laptop using 5 pin micro USB cable. W55RP20-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2 require a USB Type-C cable.

4.3 Step 3: Setup AZURE prov X.509 Example

To test the AZURE prov X.509 example, minor settings shall be done in code.

1. Setup SPI port and pin in 'w5x00_spi.h' in 'WIZnet-PICO-AZURE-C/port/ioLibrary_Driver/' directory.

Setup the SPI interface you use.

- If you use the W5100S-EVB-Pico, W5500-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2,

```
/* SPI */
#define SPI_PORT spi0

#define PIN_SCK 18
#define PIN_MOSI 19
#define PIN_MISO 16
#define PIN_CS 17
#define PIN_RST 20
```

- If you want to test with the AZURE prov X.509 example using SPI DMA, uncomment USE_SPI_DMA.

```
/* Use SPI DMA */
//#define USE_SPI_DMA // if you want to use SPI DMA, uncomment.
```

- If you use the W55RP20-EVB-Pico,

```
/* SPI */
#define USE_SPI_PIO

#define PIN_SCK 21
#define PIN_MOSI 23
#define PIN_MISO 22
#define PIN_CS 20
#define PIN_RST 25
```

2. In 'WIZnet-PICO-AZURE-C/examples/main.c', uncomment `APP_PROV_X509` to choose the sample application.

```
(...)

// The application you wish to use should be uncommented
//
//#define APP_TELEMETRY
//#define APP_C2D
//#define APP_CLI_X509
#define APP_PROV_X509
```

3. Setup network configuration such as IP in ' main.c', which is the AZURE prov X.509 example in 'WIZnet-PICO-AZURE-C/examples/' directory.

- Setup IP, other network settings to suit your network environment.

```
// The application you wish to use DHCP mode should be uncommented
#define _DHCP
static wiz_NetInfo g_net_info =
    {
        .mac = {0x00, 0x08, 0xDC, 0x12, 0x34, 0x56}, // MAC address
        .ip = {192, 168, 11, 2},                     // IP address
        .sn = {255, 255, 255, 0},                    // Subnet Mask
        .gw = {192, 168, 11, 1},                     // Gateway
        .dns = {8, 8, 8, 8},                         // DNS server
#ifdef _DHCP
        .dhcp = NETINFO_DHCP // DHCP enable/disable
#else
        // this example uses static IP
        .dhcp = NETINFO_STATIC
#endif
};
```

4.4 Step 4: Setup Device self-signed certificates

Please follow up tutorial-x509-self-sign.

1. For your reference, prepare example log as below:

   Notice! device ID = **"W5100S_EVB_PICO_PROV_X509"**

- At last stage, you need to run the following command for making .pem file.

```
MINGW64 ~/certi
$ openssl genpkey -out prov_device1.key -algorithm RSA -pkeyopt
rsa_keygen_bits:2048
...........+++++
.................................+++++

MINGW64 ~/certi
$ openssl req -new -key prov_device1.key -out prov_device1.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a
DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:.
State or Province Name (full name) [Some-State]:.
Locality Name (eg, city) []:.
Organization Name (eg, company) [Internet Widgits Pty Ltd]:.
Organizational Unit Name (eg, section) []:.
Common Name (e.g. server FQDN or YOUR name) []:W5100S_EVB_PICO_PROV_X509
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:

MINGW64 ~/certi
$ openssl x509 -req -days 365 -in prov_device1.csr -signkey
prov_device1.key -out prov_device1.crt
Signature ok
subject=CN = W5100S_EVB_PICO_PROV_X509
Getting Private key

MINGW64 ~/certi
$ openssl x509 -in prov_device1.crt -out prov_device1.pem -outform PEM

MINGW64 ~/certi
$ ls
prov_device1.crt  prov_device1.key prov_device1.csr  prov_device1.pem

MINGW64 ~/certi
$
```

2. Get the key value from files (prov_device1.crt, prov_device1.key) as below: edit ‘WIZnet-PICO-AZURE-C/exmaples/sample_certs.c’ with generated certificates as upper.

- For common name, Use "W5100S_EVB_PICO_PROV_X509" used in key generation.
- pico_az_CERTIFICATE and pico_az_PRIVATE_KEY use key value from files (prov_device1.crt, prov_device1.key)
- ‘pico_az_id_scope' use "ID Scope" string from Step 5.
- pico_az_COMMON_NAME use "device ID" from Step 4.

```
const char pico_az_id_scope[] = "0ne00xxxx5A";

const char pico_az_COMMON_NAME[] = "W5100S_EVB_PICO_PROV_X509";

const char pico_az_CERTIFICATE[] =
"-----BEGIN CERTIFICATE-----""\n"
"MIIDrTCCApUCFG9+kOlk2I815L5XAGBX7DXNxGE+MA0GCSqGSIb3DQEBCwUAMIGS""\n"
"MQswCQYDVQQGEwJLUjEUMBIGA1UECAwLR3llb25nZ2ktZG8xFDASBgNVBAcMC1Nl""\n"
...
"AwyXH6BPlQhBylsB4J5psW9ptDNKDPwF5q9cC+UiER8nSoqo0nQkB/MFSqwpZ/t0""\n"
"F7Yi3Fh/3zOiiT3qJGbFq5hU6b+AWLFjEBf4STahhOm4""\n"
"-----END CERTIFICATE-----";

const char pico_az_PRIVATE_KEY[] =
"-----BEGIN PRIVATE KEY-----""\n"
"MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQC/+cTz9dwyx0oQ""\n"
"RZq4eefN3GV6CSAUjAEVabjwO8Oa92rxAVtNhuPuFSOQMsixfW0EwPMrtBqWJx0k""\n"
...
"DHuwsI6yH1KXJ8AhQ9N99JHM00oCxVb1whKQghatpe/+4daatxD6YEoGqypxUxGv""\n"
"NCv2+ABkemj5BI2RGP5cHHk=""\n"
"-----END PRIVATE KEY-----";
```

4.5 Step 5: Setup Azure IoT Explorer

1. Create Azure Device Provisioning service.

   [MUST] For Device Provisioning service creation, please follow up the Quickstart: Set up the IoT Hub Device Provisioning Service with the Azure portal document first.

- After creating DPS, get your "ID Scope" value.

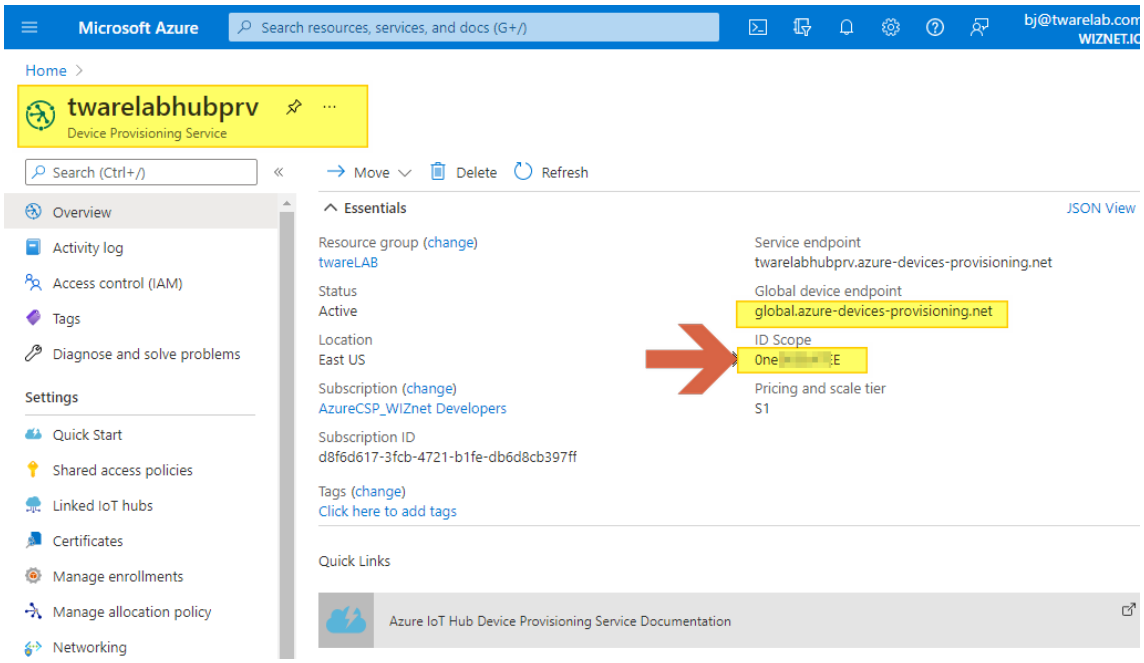

Figure 1. get "ID Scope"

2. Link to Azure IoT Hub & DPS.

- Connect DPS and IoT Hub service

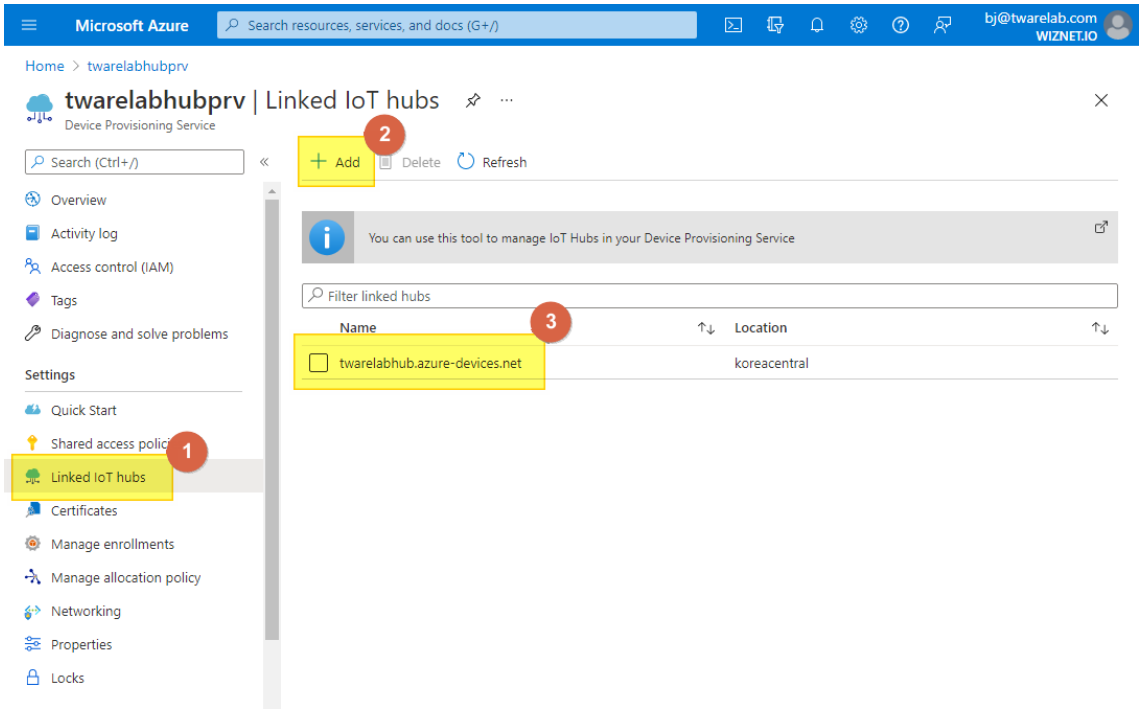

Figure 2. Linked IoT hubs

3. Create a device enrollment.

- Add individual enrollment.

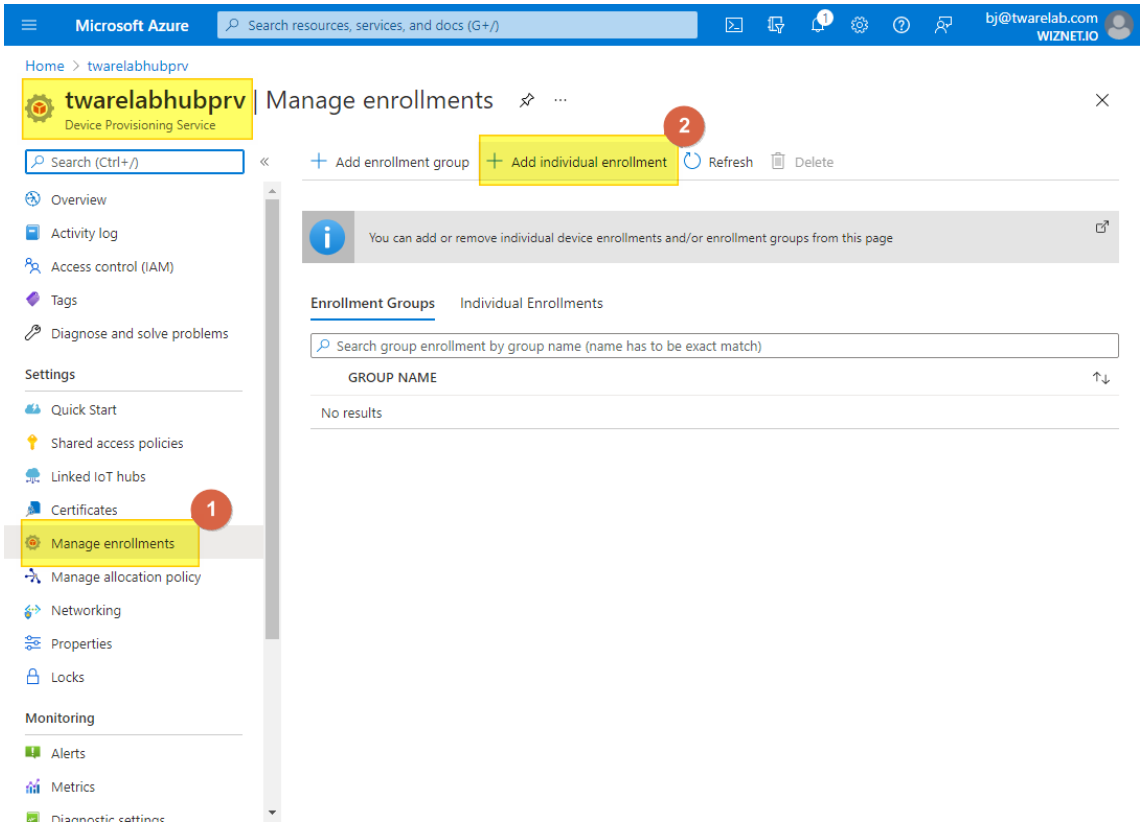

Figure 3. Manage enrollment

- Use "prov_device1.pem" file generated in previous section.

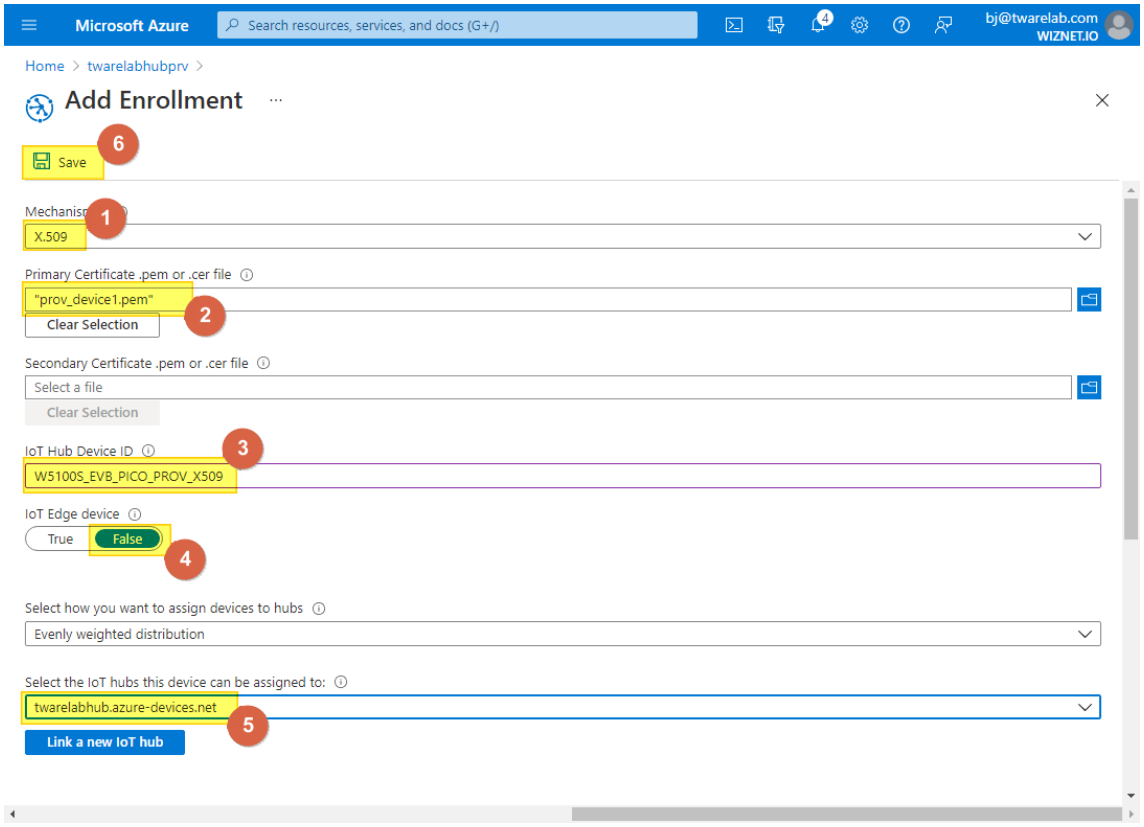

Figure 4.Enter the .pem file

- Check "Individual Enrollments" list.

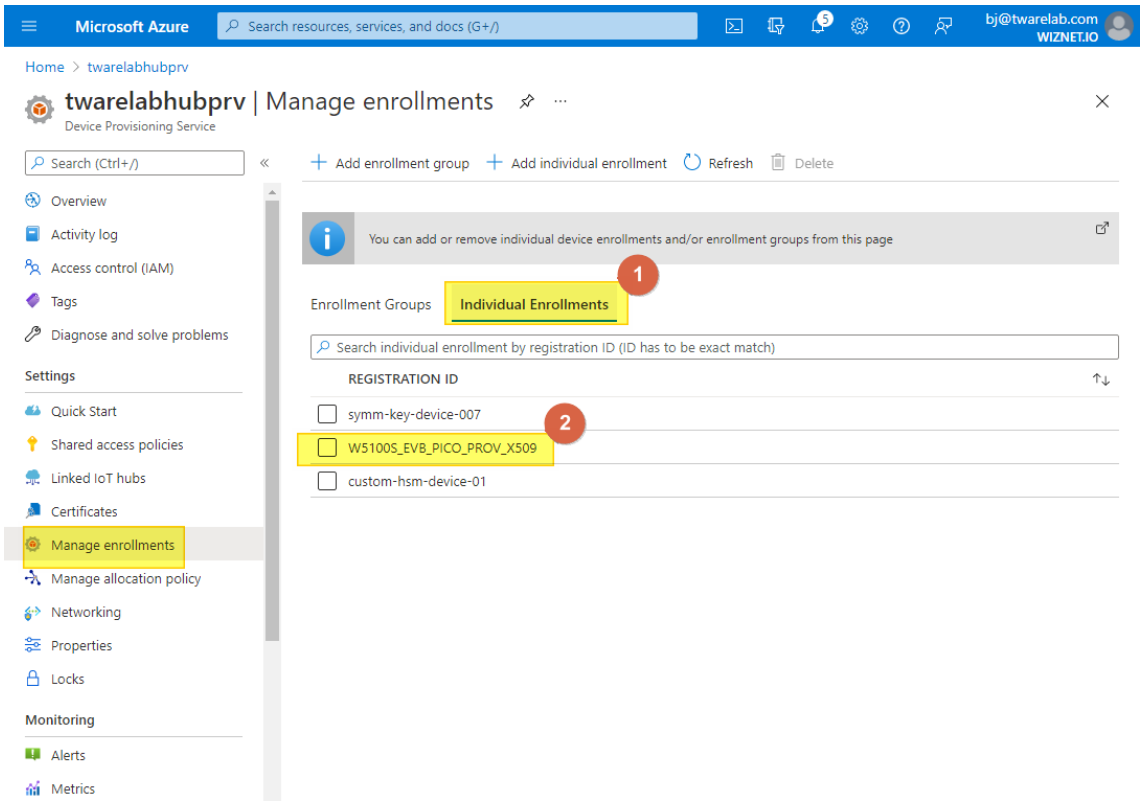

Figure 5. Individual Enrollments

- For more details,
  - Please refer How to manage device enrollments with Azure portal.
  - Or please read Quickstart: Provision an X.509 certificate simulated device document as well.

4. In the IoT Hub, go to the device menu and click "Refresh" until you see a provisioned device name.

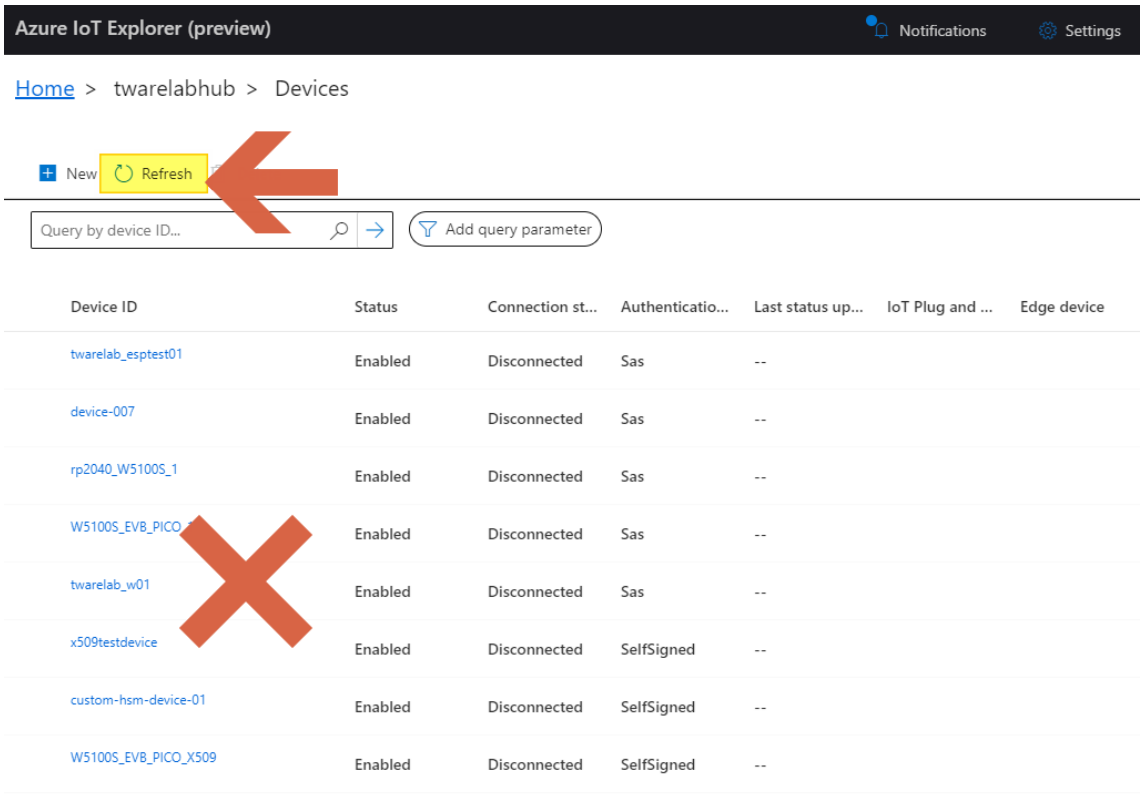

Figure 6. Click "Refresh"

- After few seconds, you can find provision device from device list as below:

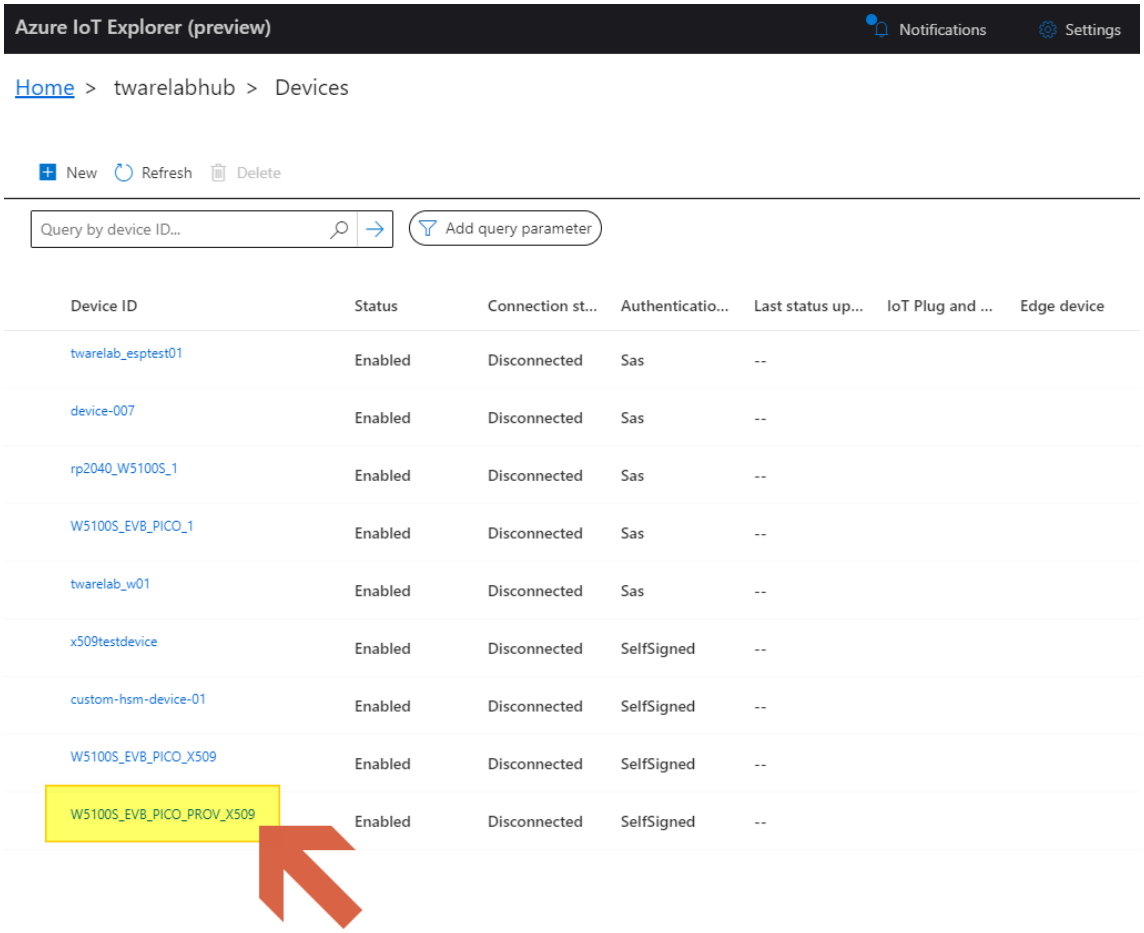

Figure 7. Click the provisioned device

5. Go to "Telemetry" menu, click "Start", and wait for incoming messages.

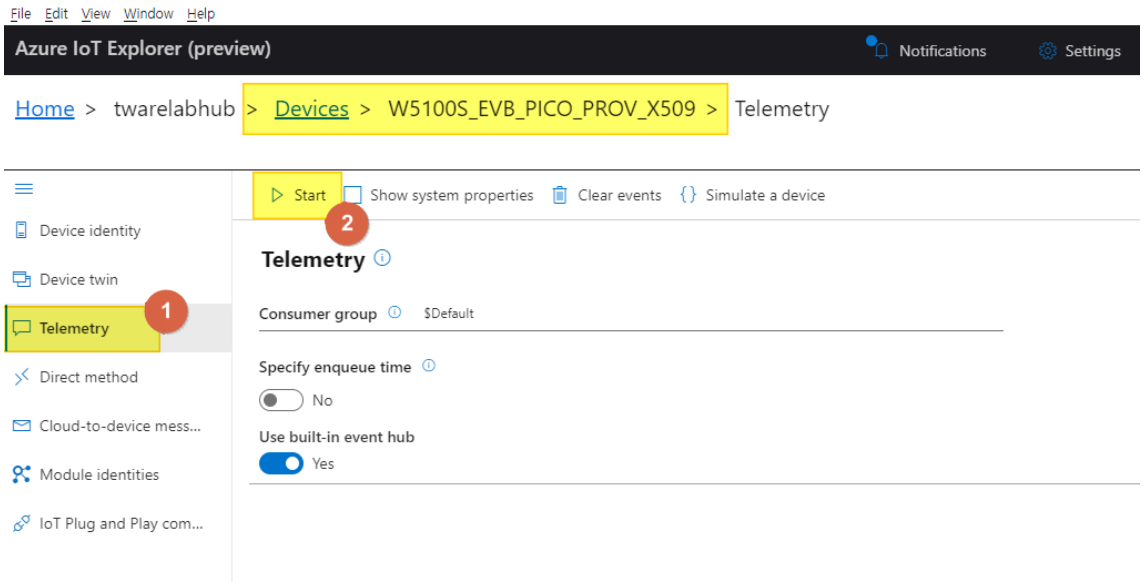

Figure 8. Click Start

4.6 Step 6: Build

1. After completing the AZURE prov X.509 example configuration, click 'build' in the status bar at the bottom of Visual Studio Code or press the 'F7' button on the keyboard to build.
2. When the build is completed, 'main.uf2' is generated in 'WIZnet-PICO-AZURE-C/build/examples/' directory.

4.7 Step 7: Upload and Run

1. While pressing the BOOTSEL button of Raspberry Pi Pico, W5100S-EVB-Pico, W5500-EVB-Pico, W55RP20-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2 power on the board, the USB mass storage 'RPI-RP2' is automatically mounted.

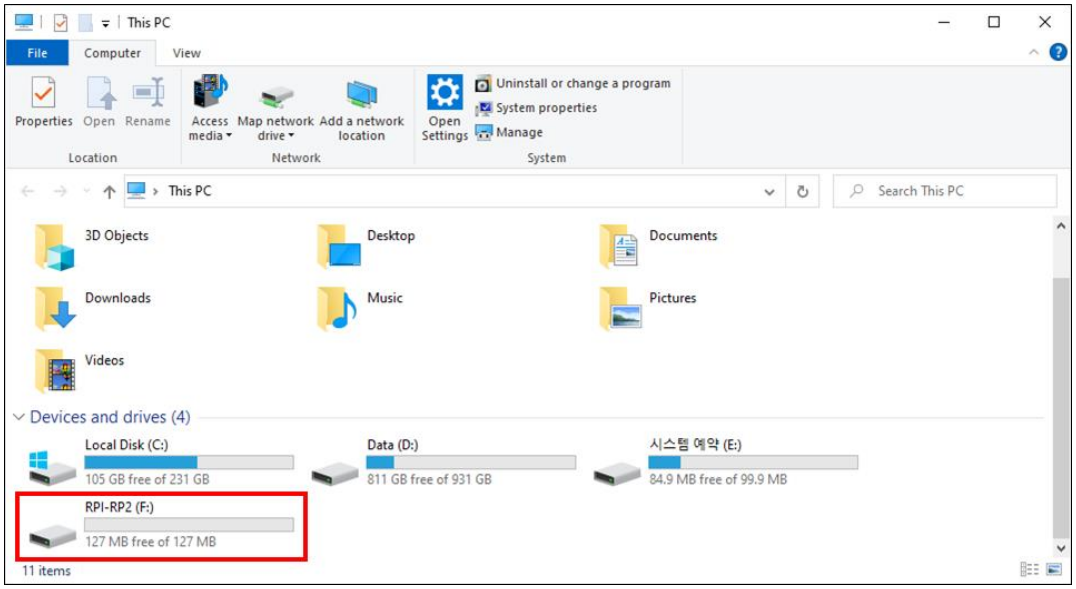

Figure 9. USB mass storage

2. Drag and drop ' main.uf2' onto the USB mass storage device 'RPI-RP2'.

3. Connect to the serial COM port of Raspberry Pi Pico, W5100S-EVB-Pico, W5500-EVB-Pico, W55RP20-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2 with Tera Term.

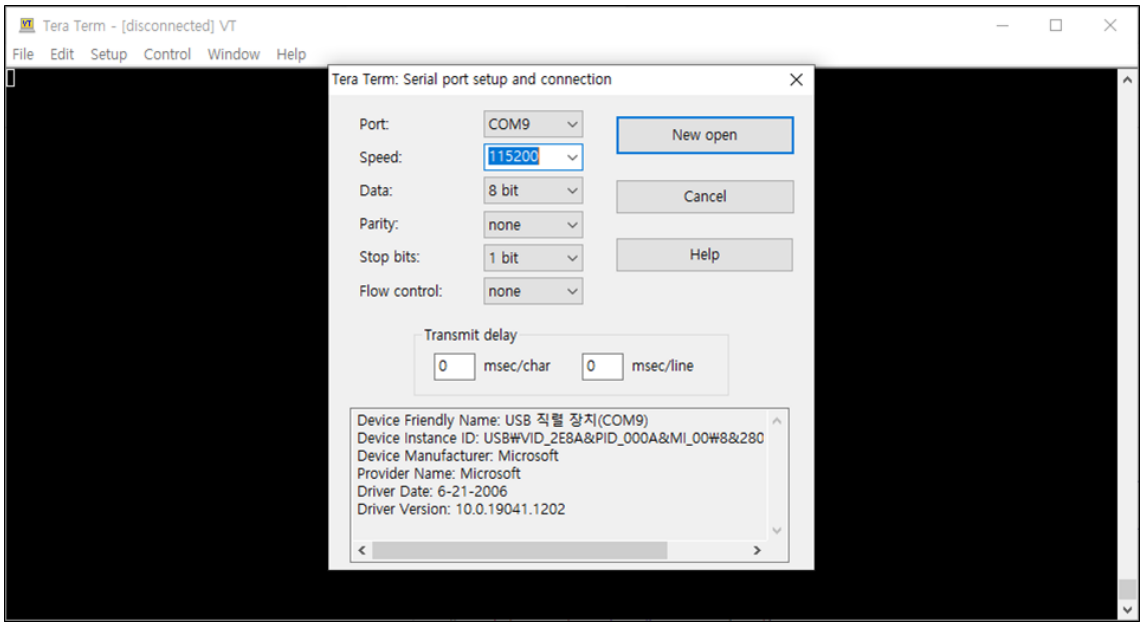

Figure 10. Tera Term

4. Reset your board.

5. If the Azure prov X.509 example works normally on Raspberry Pi Pico, W5100S-EVB-Pico, W5500-EVB-Pico, W55RP20-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2, you can see the network information of Raspberry Pi Pico, W5100S-EVB-Pico, W5500-EVB-Pico, W55RP20-EVB-Pico, W5100S-EVB-Pico2 or W5500-EVB-Pico2, Connect to Azure DPS (Device Provisioning Server) and perform the provisioning work.

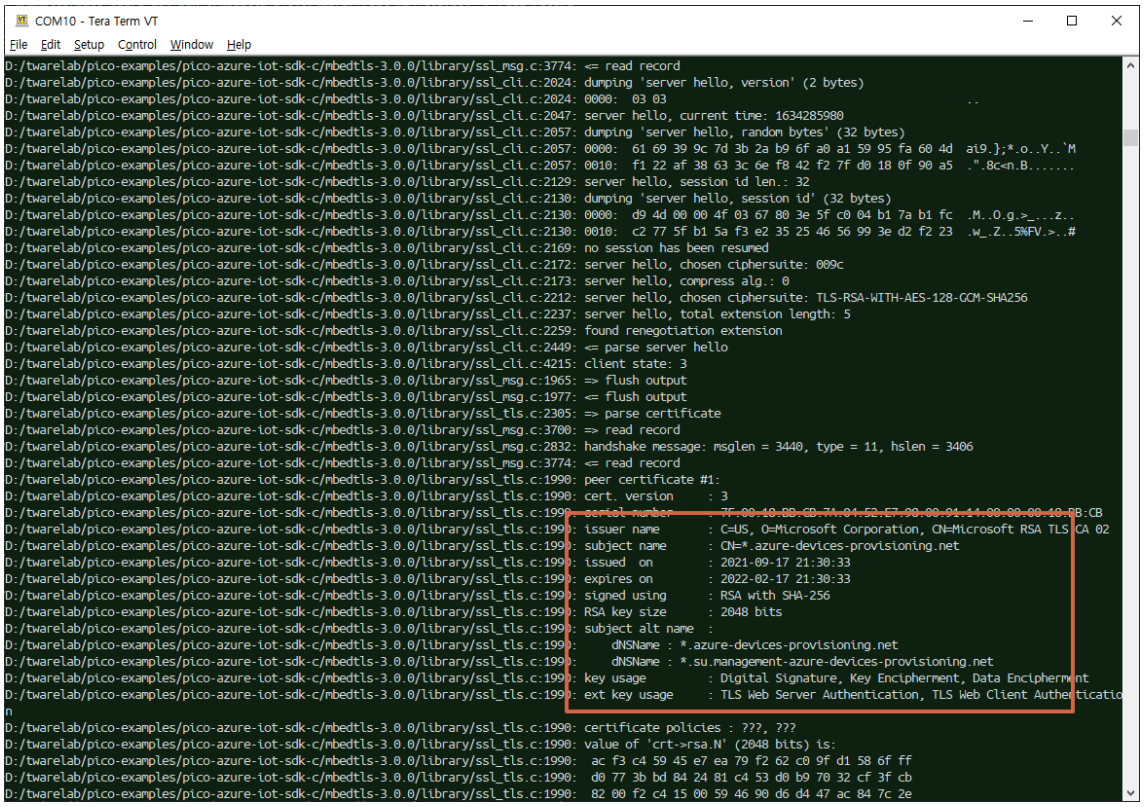

Figure 11. Connect to Azure DPS

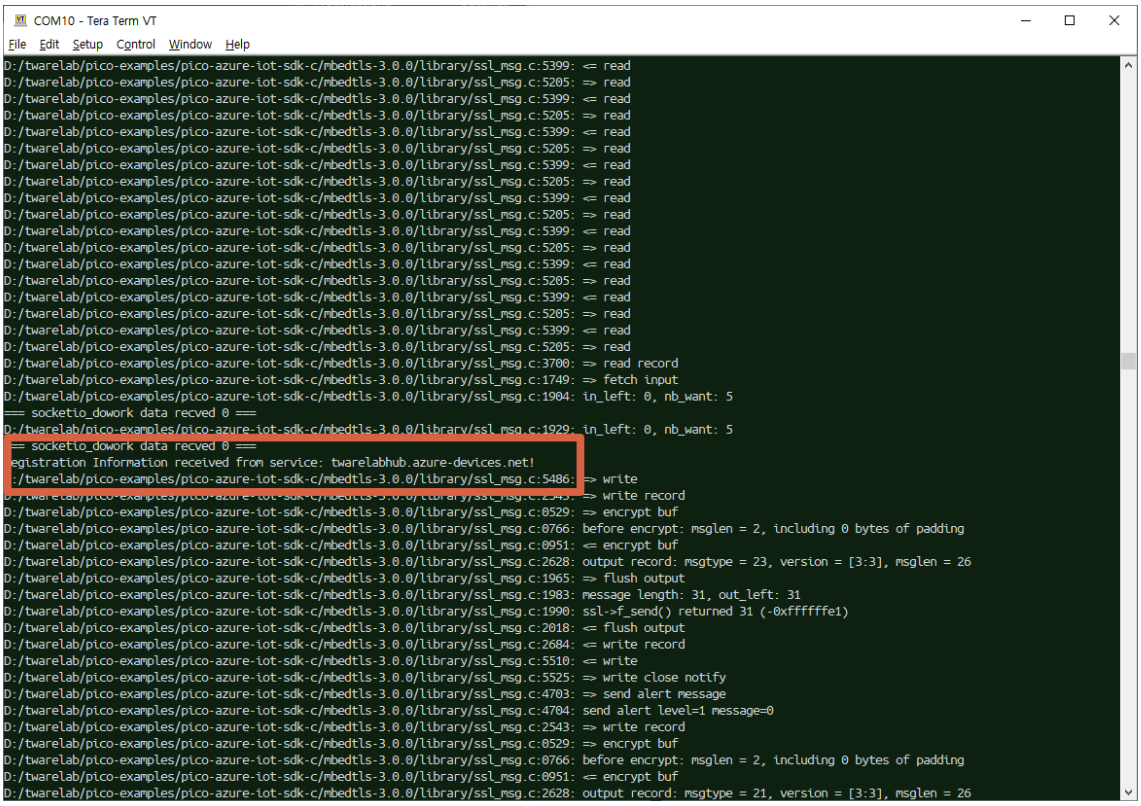

Figure 12. Provision work is done

6. After the provisioning work is complete, 2 messages are sent to Azure IoT Hub.

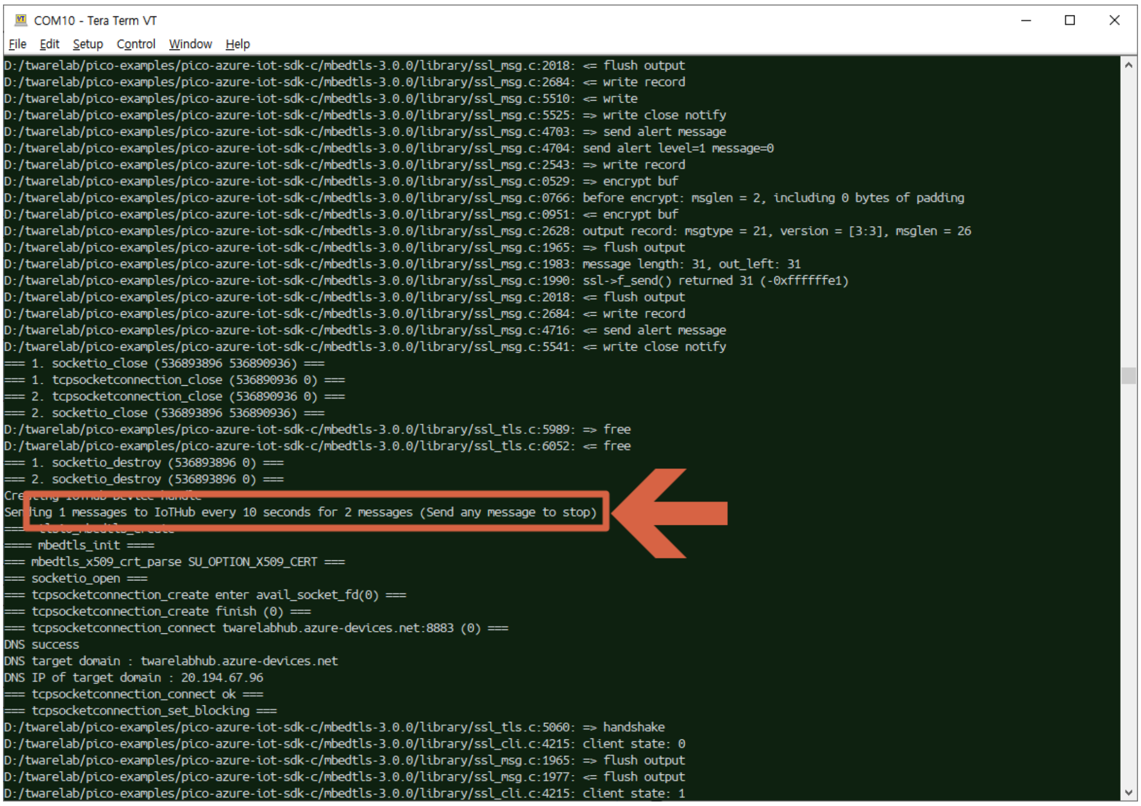

Figure 13. Send 2 messages to Azure IoT hub

7. You can see the 2 messages from the device in Azure IoT Explorer.

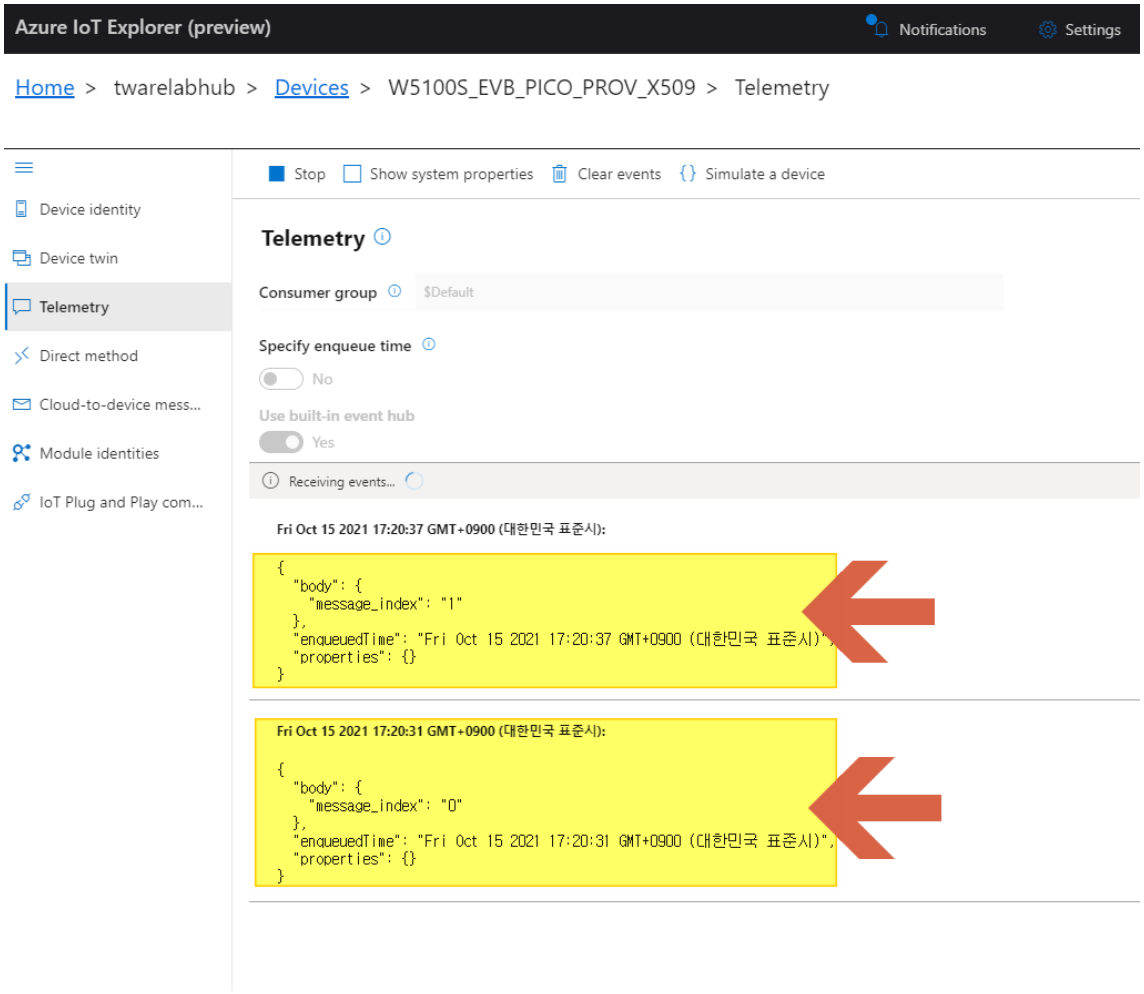

Figure 14. Receiving events in Azure IoT Explorer

Revision history

Version	Date	Descriptions
Ver. 1.0.0	Dec, 2024	Initial release.

Table 1. Revision history

Technical Support: https://forum.wiznet.io/

Sales & Distribution: sales@wiznet.io

For more information, visit our website at https://www.wiznet.io/