Ursalink Communicate with Azure&AWS

1. Topology

2. Azure/AWS configuration

Azure: https://docs.microsoft.com/en-us/azure/iot-hub/quickstart-send-telemetry-python
AWS: https://docs.aws.amazon.com/iot/latest/developerguide/iot-gs.html

3. Ursalink SDK upload

3.1 Install SD card (Prepare the SD card first), you can find the SD card slot on router as show below:

Multi Ethernet **One Ethernet**

3.2 Format SD card

Navigate to "System/General Settings/Storage"and click the "Format" button to format SD card.

3.3 Install Ursalink SDK

Navigate to "APP/Python", then Install the Ursalink SDK.

SDK installed

4. Insatall Azure/AWS SDK

Ursalink has pre-installed the SDK for you, you could check it via Putty as show below.

Note: Please contact us if you need install Azure SDK.

AWS SDK: pip install AWSIoTPythonSDK

Login as: pyuser

Password: ur123456

192.168.1.1 - PuTTY

```
login as: pyuser
pyuser@192.168.1.1's password:
*** TERMINFO:/etc/terminfo TERM:xterm *****
-- model:[redacted],sn:621380409487,hwver:0110 partnumber:L00E-S11--

---------------------------------------------------------------------------
Product Model        : [redacted]
Firmware Version     : [redacted]
---------------------------------------------------------------------------

ROUTER> en
ROUTER# conf t
ROUTER(config)# python
ROUTER(Python)#show-pylibs
-- Installed module --:
Package                    Version
-------------------------- ---------
AWSIoTPythonSDK            1.3.1
azure-iothub-device-client 1.3.5
certifi                    2018.4.16
meld3                      1.0.2
nanomsg                    1.0
paho-mqtt                  1.3.1
pip                        10.0.1
pyserial                   3.4
python-libevent            0.9.2
setuptools                 39.2.0
strict-rfc3339             0.7
supervisor                 3.3.4
ROUTER(Python)#
```

5. APP Installation

5.1 Azure test sample

Upload debug script

Navigate to "APP/Python/Python APP/Debug Script", click Browse button to upload the debug script.

5.2 Test the debug script

```
ROUTER# conf  t
ROUTER(config)# python
ROUTER(Python)#list-scripts
iothub_client_args.py    iothub_client_args.pyc   iothub_client_sample.py
ROUTER(Python)#show-pylibs
-- Installed module --:
Package                              Version
------------------------------------ ---------
AWSIoTPythonSDK                      1.3.1
azure-iot-provisioning-device-client 1.3.5
azure-iothub-device-client           1.3.5
azure-iothub-service-client          1.3.5
certifi                              2018.4.16
meld3                                1.0.2
nanomsg                              1.0
paho-mqtt                            1.3.1
pip                                  10.0.1
pyserial                             3.4
python-libevent                      0.9.2
setuptools                           39.2.0
strict-rfc3339                       0.7
supervisor                           3.3.4
ROUTER(Python)#debug iothub_client_sample.py

Python 2.7.13 (default, Jun 15 2018, 16:08:11)
[GCC 5.4.0]
IoT Hub Client for Python
'Error: Hostname not found, not a valid connection string'
Usage: iothub_client_sample.py -p <protocol> -c <connectionstring>
    protocol        : <amqp, amqp_ws, http, mqtt, mqtt_ws>
    connectionstring: <HostName=<host_name>;DeviceId=<device_id>;SharedAccessKey
=<device_key>>
ROUTER(Python)#debug iothub_client_sample.py -p mqtt

Python 2.7.13 (default, Jun 15 2018, 16:08:11)
[GCC 5.4.0]
IoT Hub Client for Python
'Error: Hostname not found, not a valid connection string'
Usage: iothub_client_sample.py -p <protocol> -c <connectionstring>
    protocol        : <amqp, amqp_ws, http, mqtt, mqtt_ws>
    connectionstring: <HostName=<host_name>;DeviceId=<device_id>;SharedAccessKey
=<device_key>>
ROUTER(Python)#
```

5.3 In the following sections, AWS is used as an example

"Serial port Utility" simulate Terminal equipment with RS232 that connect to Ursalink.

5.4 Import APP

Navigate to "APP/Python/Python APP", click the "Browse" button, select the Ursalink SDK and click Import button.

Note: When you use the demo APP, you need to reconfigure the "config.ini" and also replace the certs to yours. APP should be with ZIP format.

5.5 Enable APP

Navigate to “ APP/Python/AppManager Configuration/” and Enable AppManager.

5.6 AppManager Status

Click the “View” button to check more details of the App.

User name : pyapp password: py123456app

5.7 Test AWS

In the AWS IoT console, in the left navigation pane, choose Test.

On the MQTT client page, in the Publish section, in the Specify a topic and a message to publish… field, type my/topic or the topic you used in the rule.