

## Advantech AE Technical Share Document

<b>Date</b>	2018 / 11 / 8	<b>SR#</b>	
<b>Category</b>	■FAQ □SOP	<b>Related OS</b>	N/A
<b>Abstract</b>	How to set the authentication function of ADAM MQTT?		
<b>Keyword</b>	ADAM-60XX, ADAM-62XX, MQTT, authentication, username, password		
<b>Related Product</b>	ADAM-6217-B, ADAM-6017-D ADAM-6050-D, ADAM-6051-D, ADAM-6052-D, ADAM-6060-D, ADAM-6066-D ADAM-6250-B, ADAM-6251-B ,ADAM-6256-B, ADAM-6260-B, ADAM-6266-B		

### ■ Problem Description:

This document explains how to set the authentication function of ADAM MQTT for connecting to the broker that requires username and password for verification.

### ■ Answer:

Below is the support table of ADAM MQTT authentication function. User need to check module HW version and upgrade to certain FW version for setting the authentication of MQTT.

DIO Model	FW version	AIO Model	FW version
ADAM-6050/51/52/60/66-D	After v6.02 B01	ADAM-6017-D	After v6.02 B00
ADAM-6250/51/56/60/66-B	After v6.02 B01	ADAM-6217-B	After v6.02 B01

After checking the HW, FW, Utility version (after 2.05.11B17) all support MQTT authentication function, we can start the DEMO. Following are the steps by steps (with ADAM-6250) to test the MQTT function with username/password:

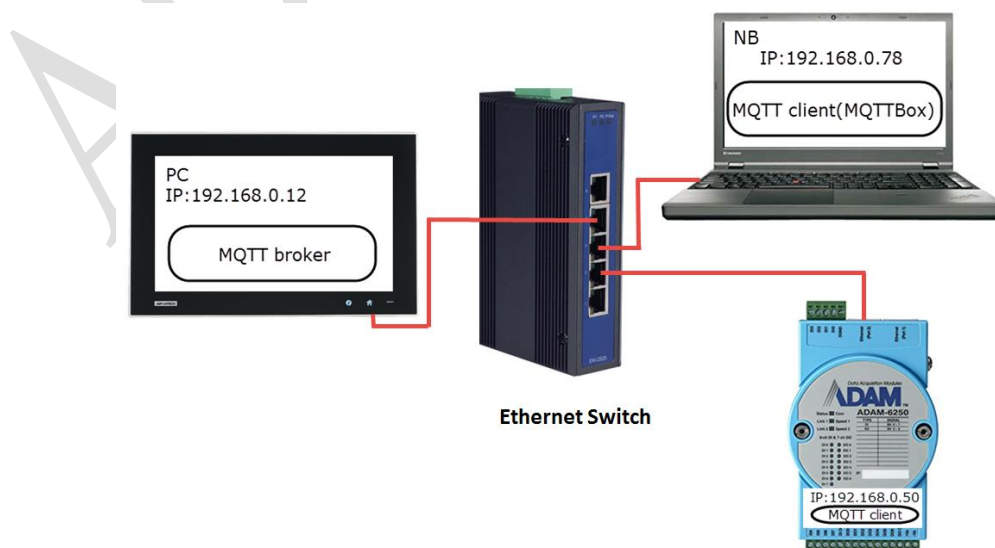
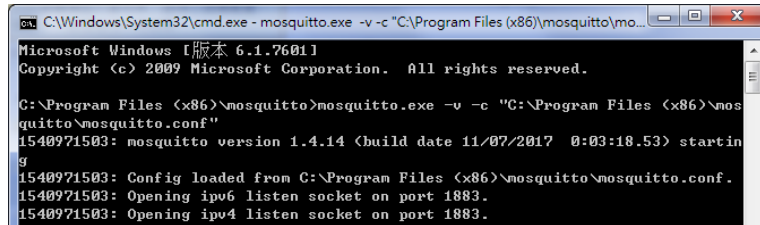


Figure1. Application structure

## 1. Build up a mosquitto broker with username/password authentication and run it.

First, in order to test the username/password function, we have to build up a broker which can only be accessed with correct username/password. We can use Webaccess or Mosquitto to build up a MQTT broker with authentication. In this example I use Mosquitto. (For the Webaccess MQTT broker setting, you can refer to the end of this document.)



```

C:\Windows\System32\cmd.exe - mosquitto.exe -v -c "C:\Program Files (x86)\mosquitto\mosquitto.conf"
Microsoft Windows [版本 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Program Files (x86)\mosquitto>mosquitto.exe -v -c "C:\Program Files (x86)\mosquitto\mosquitto.conf"
1540971503: mosquitto version 1.4.14 (build date 11/07/2017 0:03:18.53) starting
1540971503: Config loaded from C:\Program Files (x86)\mosquitto\mosquitto.conf.
1540971503: Opening ipv6 listen socket on port 1883.
1540971503: Opening ipv4 listen socket on port 1883.
  
```

Figure2. Mosquitto broker (username:jjj/password:123) which run on IP:192.168.0.12.

## 2. Use Adam/Apax .NET Utility to setup the MQTT setting.

We have to setup the ADAM module. There are several parameters we have to fill in (Figure3). After pressing the “Apply” button, we can see the module successfully connect to the broker not only on the broker page (Figure4) but also in the wireshark. (We can see the module connect to the broker with correct username/password in Figure5.)



Information | Network | Stream/Trap | Administration | Firmware | P2P/Event | Access Control | Modbus Address | Cloud

MQTT | SNTP

Publish / Subscribe : ☒ Enable Apply

Host : 192.168.0.12

UserName / Password : jjj / 123

Heartbeat / Deadband : 5 second(s) / 100 milli-second(s)

Retain Message : ☒ Enable

Will Topic : Advantech/00D0C9FE962E/Device\_Status

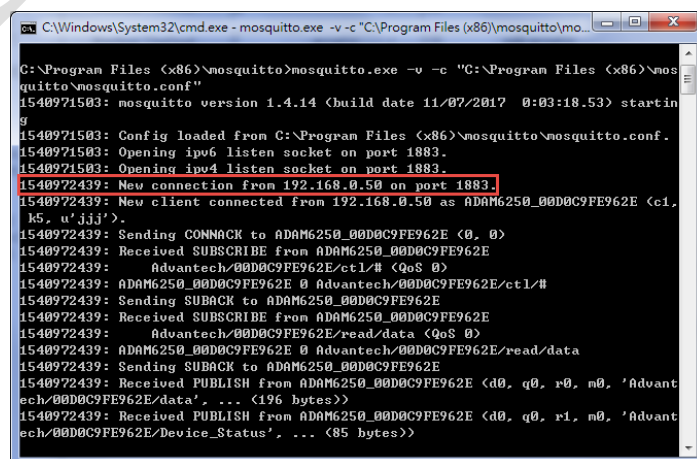
Publish QoS : 0

Publish Topic : Advantech/00D0C9FE962E/data

Subscribe QoS : 0

Subscribe Topic : Advantech/00D0C9FE962E/ctl/do1  
Advantech/00D0C9FE962E/ctl/do2  
Advantech/00D0C9FE962E/ctl/do3  
Advantech/00D0C9FE962E/ctl/do4  
Advantech/00D0C9FE962E/ctl/do5  
Advantech/00D0C9FE962E/ctl/do6  
Advantech/00D0C9FE962E/ctl/do7

Figure3. Utility setting (under Cloud/MQTT tab)



```

C:\Program Files (x86)\mosquitto>mosquitto.exe -v -c "C:\Program Files (x86)\mosquitto\mosquitto.conf"
1540971503: mosquitto version 1.4.14 (build date 11/07/2017 0:03:18.53) starting
1540971503: Config loaded from C:\Program Files (x86)\mosquitto\mosquitto.conf.
1540971503: Opening ipv6 listen socket on port 1883.
1540971503: Opening ipv4 listen socket on port 1883.
1540972439: New connection from 192.168.0.50 on port 1883.
1540972439: New client connected from 192.168.0.50 as ADAM6250_00D0C9FE962E (c1, k5, u'jjj').
1540972439: Sending CONNACK to ADAM6250_00D0C9FE962E (0, 0)
1540972439: Received SUBSCRIBE from ADAM6250_00D0C9FE962E
1540972439: Advantech/00D0C9FE962E/ctl/# (QoS 0)
1540972439: ADAM6250_00D0C9FE962E 0 Advantech/00D0C9FE962E/ctl/#
1540972439: Sending SUBACK to ADAM6250_00D0C9FE962E
1540972439: Received SUBSCRIBE from ADAM6250_00D0C9FE962E
1540972439: Advantech/00D0C9FE962E/read/data (QoS 0)
1540972439: ADAM6250_00D0C9FE962E 0 Advantech/00D0C9FE962E/read/data
1540972439: Sending SUBACK to ADAM6250_00D0C9FE962E
1540972439: Received PUBLISH from ADAM6250_00D0C9FE962E (d0, q0, r0, m0, 'Advantech/00D0C9FE962E/data', ... (196 bytes))
1540972439: Received PUBLISH from ADAM6250_00D0C9FE962E (d0, q0, r1, m0, 'Advantech/00D0C9FE962E/Device_Status', ... (85 bytes))
  
```

Figure4. ADAM-6250 connects to the broker.

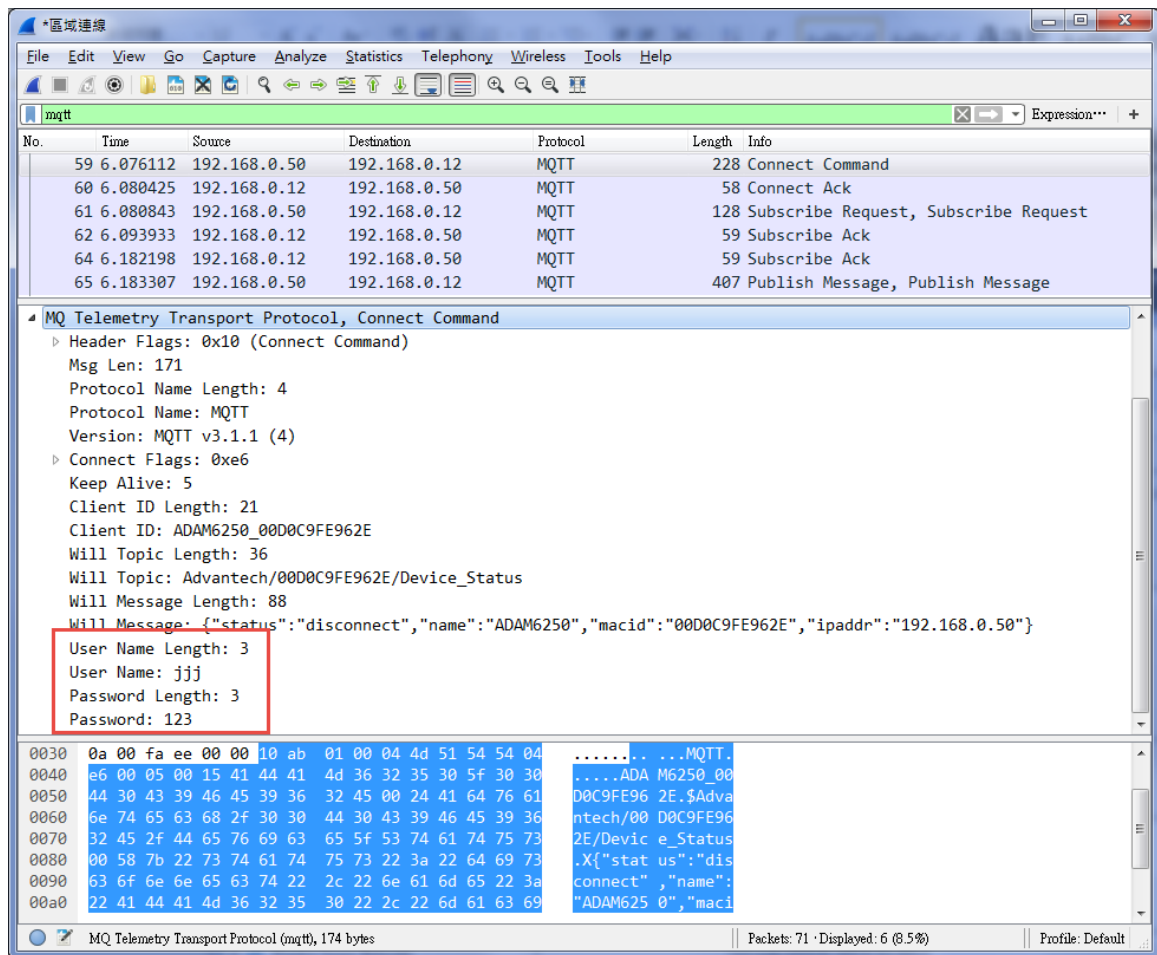


Figure5. Connect packet

### 3. Use a 3<sup>rd</sup> party MQTT client (MQTTBox) to get the DI status.

In Figure6, we have to setup a MQTT client for testing the function. In Figure7, enter the right Topic and press Subscribe button, you can get the IO data. In Figure8, enter the right Topic and payload then press Publish button, you can set the DO. You can also observe the DOO change in the right hand side (Subscribe block) of this figure.

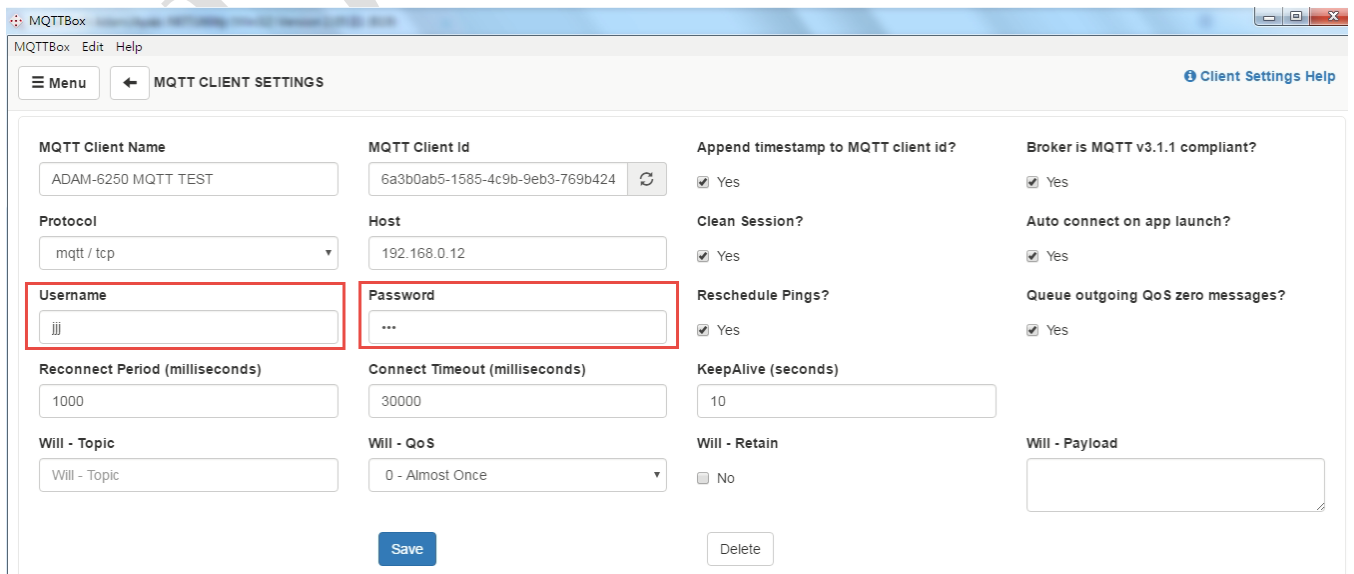


Figure6. MQTTBox setting page

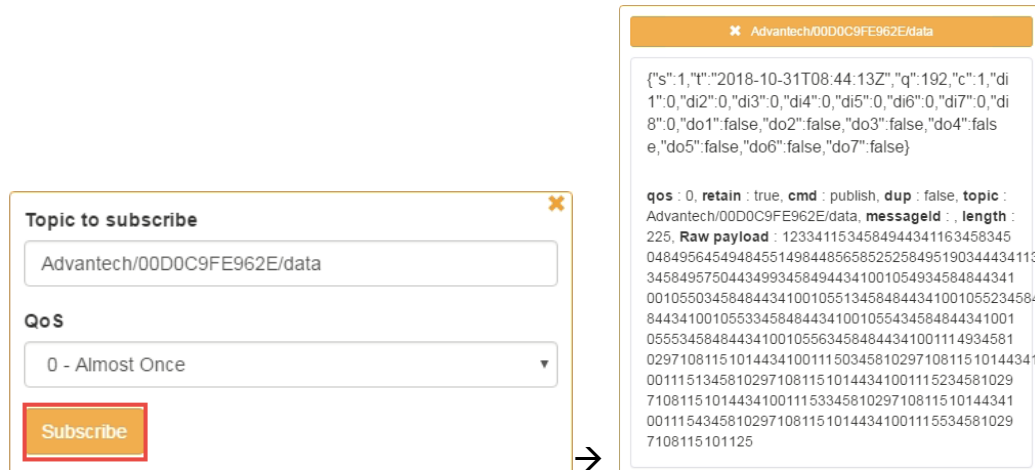


Figure7. Subscribe IO data

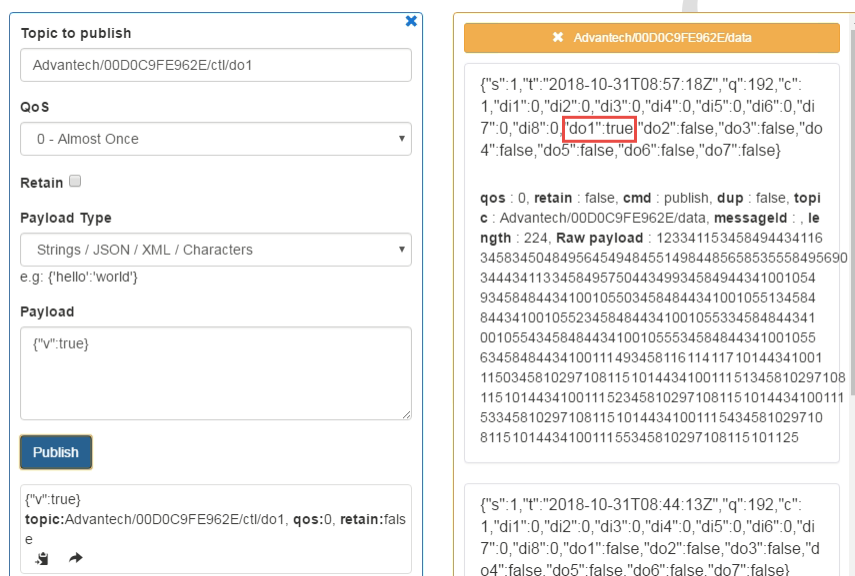


Figure8. Publish DO data

## Remarks:

- Utility only allows user enter max 20 characters for username and password. However, module can support 49 characters for username and 99 characters for password for the MQTT authentication function. If users need to set longer characters for both username and password, they can use ASCII command to configure this setting.

%aaSETMQTTUNxx...x	Set MQTT user name aa: always 01 xx...x: user name, if set null module will disable the user- name and password func- tion.	Return: >01 Error: ?01
%aaSETMQTTPWxx...x	Set MQTT password aa: always 01 xx...x: password, if set null module will disable the user- name and password func- tion.	Return: >01 Error: ?01

Figure9. ASCII command for setting the username/password

2. Here is a packet about the MQTT client connect the broker with wrong username/password.

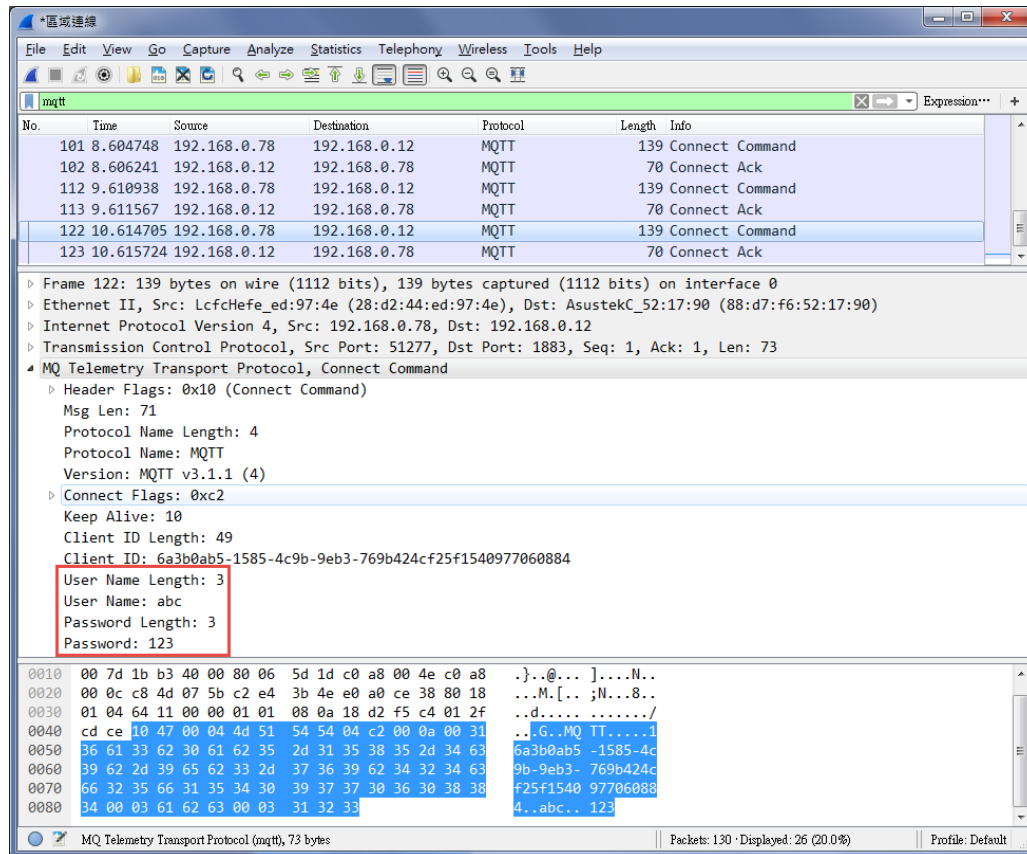


Figure10. Client connects to broker with incorrect username/password.

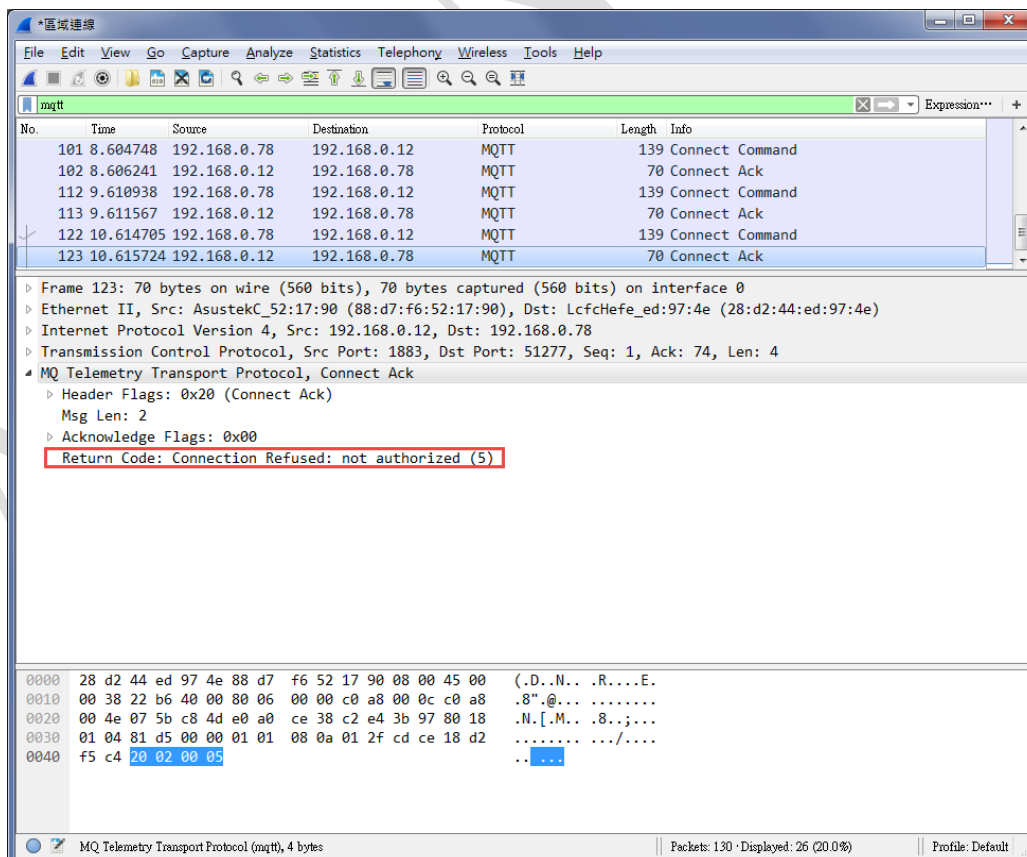


Figure11. Broker responses with a "connect act" with "not authorized".

3. Webaccess MQTT broker setup SOP (After v8.3.3)
  - a. Open the project manager and click the MQTT Broker.

**Advantech WebAccess Project Manager** [Quick Start](#) [Help](#) [Logout](#)

**Current Project(s)**

Project Name	Project	Dashboard	Description	IP	HTTP Port	TCP Port	Timeout	Update	Delete
CloudProject	<a href="#">Configure</a>	<a href="#">Edit</a>	Project Description	PC060607	0	4592	0	<a href="#">Update</a>	<a href="#">Delete</a>
demo	<a href="#">Configure</a>	<a href="#">Edit</a>	Project Description	127.0.0.1	0	0	0	<a href="#">Update</a>	<a href="#">Delete</a>
WISE2410	<a href="#">Configure</a>	<a href="#">Edit</a>	Suzhou Summit 2018	127.0.0.1	0	4592	0	<a href="#">Update</a>	<a href="#">Delete</a>

Please select one of above available Projects to start!!

[Integrity Checking](#)
[Backup](#)
[Restore](#)
[Admin/Project User](#)
[ODBC Log Data Source](#)
[WebAccess Express](#)
[Dashboard Settings](#)
[Setup HTTPS Service](#)
[MQTT Broker](#)
[System Log](#)
[Action Log](#)
[Alarm Log](#)
[Analog Tag Log](#)
[Analog Change Log](#)
[Discrete Tag Log](#)
[Text Tag Log](#)
[Event Log](#)
[LogData Maintenance](#)

**Project Configuration**

**Create New Project**

Project Name

Project Description

Project Node IP Address

Project Node HTTP Port

Project Primary TCP Port

Project Timeout

Remote Access Code

Retype Remote Access Code

Log Changes to System Log ☐ Yes ☒ No

- b. Enter the username/password you want and press “Submit” button.

**MQTT Broker**

**MQTT Broker Settings**

Enable ☒ Yes ☐ No

UserName

Password  ☐ Show Password

TCP Port

TLS Port

Websocket Port

Websocket TLS Port

Project Node Public IP

[\[Cancel\]](#)

- c. Done!