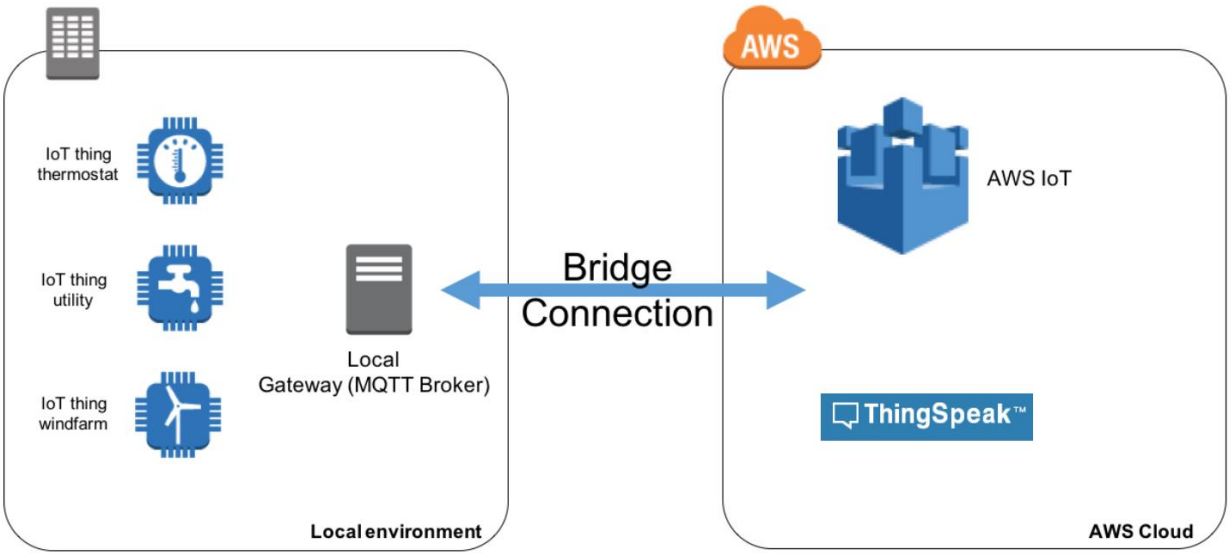


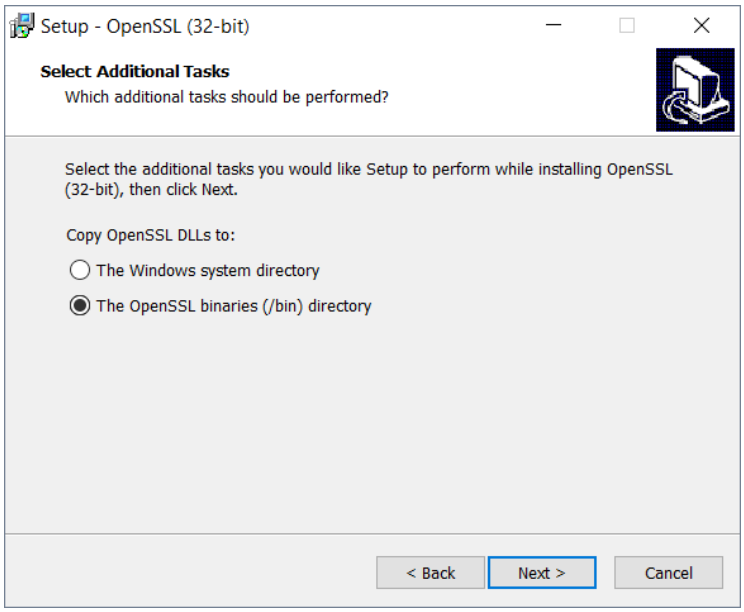
Fig 1. General concept (adopted from Amazon page)

Installation

1) Mosquitto (MQTT broker)

There are several step:

- Install OpenSSL (Win32OpenSSL-1_0_2n.exe) to root folder
  - On step “Select Additional Tasks” choose copy DLLs to /bin directory



- Installing mosquitto (mosquitto-1.4.15a-install-win32.exe)
- Put pthreadVC2.lib in mosquitto folder

2) Beckhoff side

Install TwinCAT (link to download last release)

Requirements on all devices at least TwinCAT v.3.1.4022

Turning mosquitto broker on

- Execute mosquito.exe from CLI
  - Open cmd on Windows (press win + r, and type cmd)
  - Navigate to folder where mosquito was installed

    Example:

```
C:\>cd "Program Files (x86)\mosquitto"
```

  - mosquitto.exe (the command will start MQTT broker with default configurations, more detailed information on configuration file could be found here)
  - To start the broker with custom configuration file execute mosquitto with –c attribute as follows:

```
C:\Program Files (x86)\mosquitto>mosquitto.exe -c C:\somePathtoConfFile\configurationFile.conf
```

- To start broker in bridge mode you have to choose one of .conf files
  - thingSpeakBridgeTLS.conf
    - bridging all topics which start with remote/thingspeak to ThingSpeak cloud
  - awsBridgeTLS.conf
    - bridging localgateway_to_awsiot topic to AWS IoT cloud
- In .conf files several changes must be made
  - Path to certificates have to be specified

For ThingSpeak:

```
# ================================================================
# Certificate based SSL/TLS support
# ----------------------------------------------------------------
# Path to the rootCA
# ###### has to be changed to specific path to ca-certificates.crt file

bridge_cafile ######\mosquittoConfFiles\thingSpeakCerts\ca-certificates.crt
```

For AWS:

```
# ================================================================
# Certificate based SSL/TLS support
# ----------------------------------------------------------------
# certificates must be generated in your aws console and loaded to
# mosquittoConfFiles\awsCerts directory, ###### change to path on your machine

# Path to the rootCA
bridge_cafile ######\mosquittoConfFiles\awsCerts\rootCA.pem

# Path to the PEM encoded client certificate
bridge_certfile ######\mosquittoConfFiles\awsCerts\cert.crt

# Path to the PEM encoded client private key
bridge_keyfile ######\mosquittoConfFiles\awsCerts\private.key
```

Troubleshooting with mosquitto broker

```
1529574829: mosquitto version 1.4.15 (build date 23/03/2018 10:26:57.31) starting
1529574829: Config loaded from Z:\h18202\Documents\IoTproject\IoTProjectSrc\mosquittoConfFiles\thingSpeakBridgeTLS.conf.

1529574829: Opening ipv6 listen socket on port 1883.
1529574829: Error: Only one usage of each socket address (protocol/network address/port) is normally permitted.
```

If you are facing problem with opening socket on port, it means that instance of mosquitto has been already launched and you will have to stop it manually by executing following:

sc stop mosquitto

Note: Command prompt has to be run as administrator

TwinCAT project

Beckhoff has a sample, which works out of a box with several changes, that example also can be found in IoTProjectSrc directory. The example send value of counter, which goes up and down, each 15 seconds. Before running the project several changes has to be done in PrgMqttCom function file:

- Api key and channel id have to be changed to your own
- Address of host (in our case broker's one) have to be changed
- Topics should to be specified for your needs

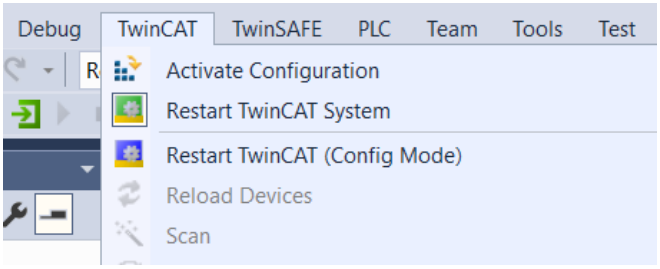
PrgMqttCom | MAIN | Library Manager

```
1  PROGRAM PrgMqttCom
2  VAR
3      fbMqttClient : FB_IotMqttClient;
4      bSetParameter : BOOL := TRUE ;
5      bConnect : BOOL := TRUE ;
6
7      (* published message *)
8      sTopicPub : STRING(255) := 'remote/thingspeak/channelID/publish/writeAPIkey';
9      // from thingspeak channel profile
10     sPayloadPub : STRING(255);
11     i: UDINT := 0 ;
12     switch : BOOL := TRUE;
13     fbTimer : TON := (PT:=T#15S);
14
15     (* received message *)
16     (*bSubscribed : BOOL;
17     sTopicSub : STRING(255) := 'test';
```

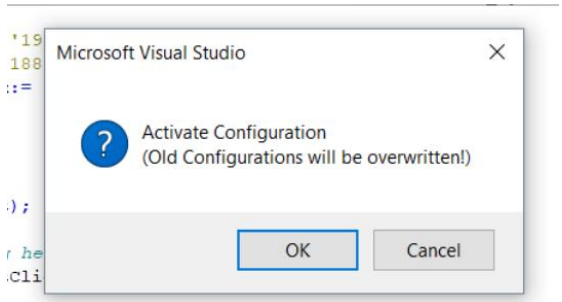
```
1  IF bSetParameter THEN
2      fbMqttClient.sHostName:= '192.168.0.40'; // address of host
3      fbMqttClient.nHostPort:= 1883; // port which bridge are listening to
4      fbMqttClient.sTopicPrefix:= '';
5      bSetParameter := FALSE;
6  END_IF
```

To run project:

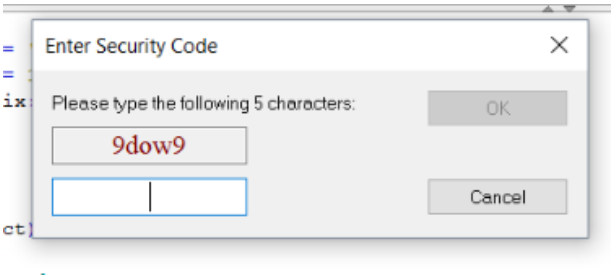
1) Activate Configuration in order to configure project install required licenses



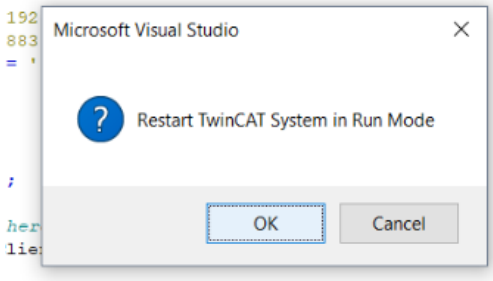
Hit “OK” to proceed configuration



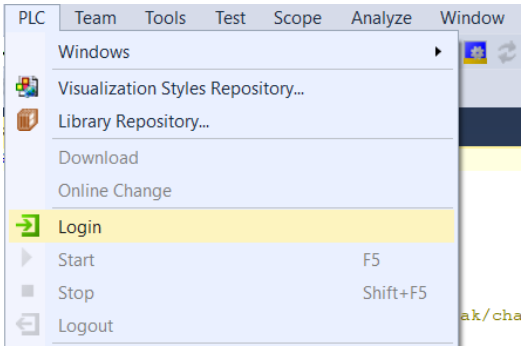
It could ask to type security code for updating required licenses



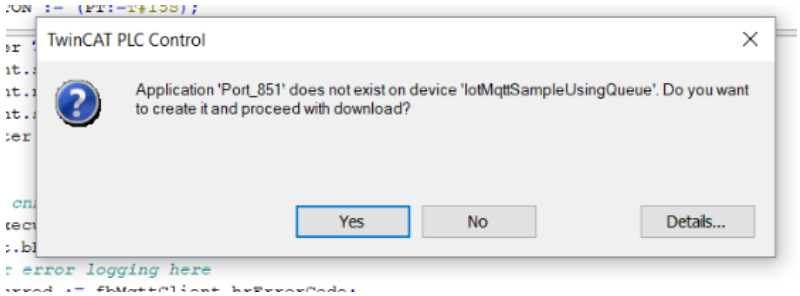
Press “OK” to restart/start TwinCAT in run mode



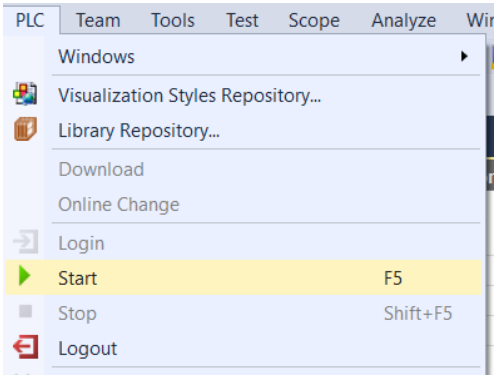
2) Login and run compiled project
In PLC tab press Login button or on front-page toolbar panel



Press “Yes”



Finally, press “Start” button in PLC tab or on the toolbar panel or F5



To stop running program press “Stop” in PLC tab or Shift + F5 and then “Logout”

After starting program, logs of sending data can be found in broker window.

To test settings of cloud side or debug work of broker and program, mosquitto provides two utilities mosquitto_pub and mosquitto_sub, detailed information about them can be find here and here. The folder of the utilities is the same as for mosquitto broker.

To publish test message to ThingSpeak cloud:

mosquitto_pub.exe -h mqtt.thingspeak.com -p 1883 -t channels/chatID/publish/writeAPIKey -m "field1=100"

To subscribe to our broker and listen to all topics, all data, which is going through:

mosquitto_sub.exe -p 1883 -t #

Resources

All necessary files can be found in IoTProjectSrc.zip

Detailed information on mosquitto broker: https://mosquitto.org/

To get API keys create an account on https://thingspeak.com

To get API keys for AWS IoT follow instruction on
https://docs.aws.amazon.com/iot/latest/developerguide/iot-gs.html