

CPE 314: Computer Networks (2/2561)

Project I

MQTT (MQ Telemetry Transport) is a popular application layer protocol for Internet of Things (IoT) applications. The communication model is the published-subscribed pattern, which consists of three node roles – Publisher, Subscriber, and Broker. A publisher publishes its topic to Broker and one or more subscribers subscribe for the topic. Whenever the publisher publishes data, Broker will automatically relay it to all nodes subscribed to that topic.

A group of at most five students is to design and implement a simple command-line version of MQTT with the above functionality. So, the software will have three parts – Publisher, Broker, and Subscriber. The software can be coded in any programming language you like (Python, C, Java, etc.). You are not allowed to use any MQTT library APIs.

Software Requirements

1. The subscriber subscribes to a given topic with the command line:

```
subscribe broker_ip_address 'topic_name'
```

For example, to subscribe to the topic `/room1/light` with the broker ip address `202.44.12.85`, enter the command

```
subscribe 202.44.12.85 '/room1/light'
```

Once started, the subscriber program stays running to receive published messages and print them on the screen, and terminates when receiving Ctrl-C key or something similar from the user input.

2. The publisher publishes data to a given topic with the command line:

```
publish 'broker_ip_address' 'topic_name' 'data to publish'
```

For example,

```
publish 202.44.12.85 '/room1/light' 'value=on'
```

```
publish 10.0.3.2 '/room2/lcd' 'hello world'
```

3. The broker prints out a message received from a publisher and sends it to all subscribers for that topic. The broker must be able to handle multiple publishers and subscribers for different topics at the same time.

Deliverables

By midnight of **Sunday March 3, 2018**, submit in one `.zip` file containing the final report, the source code (and executable files if any), and the `readme` file describing how to compile and run the program.

The final report must contain the following information:

Program flowchart Explain the sequence of program operations on publisher, broker, and subscriber.

Protocol design Explain the message syntax, semantic, timing and actions of your protocol.

Implementation Explain how your design is implemented into different pieces of codes.

Demo

Sign up for demo (10 minutes per group) in allocated time slots (TBA) during 11-15 March, 2019. One person in a group will be randomly chosen to demo, explain the codes, and answer questions.

The grading is based on design explanation (35%) program completeness (35%), report quality and writing (10%), and presentation (20%).

All parts of the project must be your own group's work. Any form of copying from another group/source is an academic fraud and may receive zero credits as the highest penalty.