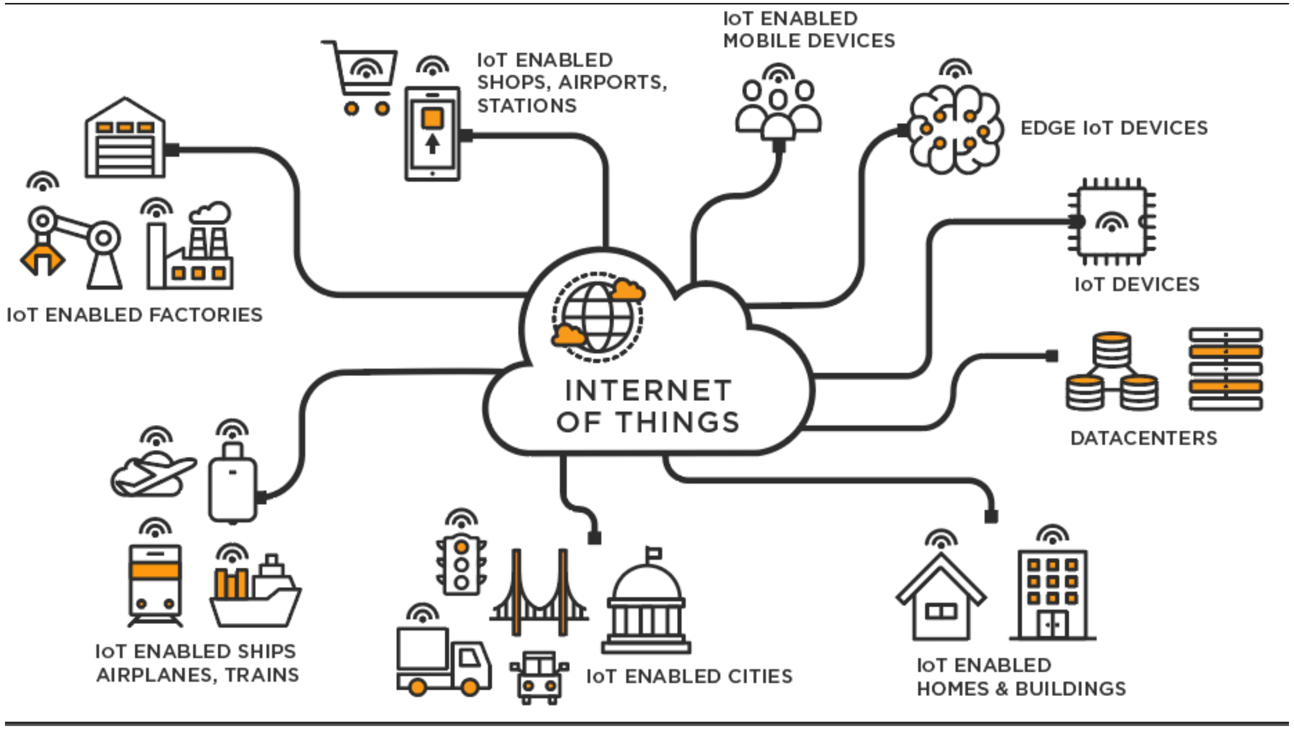

IoT ENABLED SHOPS, AIRPORTS, STATIONS
IoT ENABLED MOBILE DEVICES
EDGE IoT DEVICES
IoT ENABLED FACTORIES
IoT DEVICES
INTERNET OF THINGS
DATACENTERS
IoT ENABLED SHIPS AIRPLANES, TRAINS
IoT ENABLED CITIES
IoT ENABLED HOMES & BUILDINGS

MQTT

Introduction

- MQTT stands for **Message Queuing Telemetry Transport**
- MQTT is a machine to machine internet of things connectivity protocol
- It is an extremely lightweight and publish-subscribe messaging transport protocol
- This protocol is useful for the connection with the remote location where the bandwidth is a premium
- These characteristics make it useful in various situations, including constant environment such as for communication machine to machine and internet of things contexts
- It is a publish and subscribe system where we can publish and receive the messages as a client
- It makes it easy for communication between multiple devices
- It is a simple messaging protocol designed for the constrained devices and with low bandwidth, so it's a perfect solution for the internet of things applications

History

- The MQTT was developed by Dr. Andy Stanford-Clark, IBM, and Arlen Nipper
- The previous versions of protocol 3.1 and 3.1.1 were made available under MQTT ORG
- In 2014, the MQTT was officially published by OASIS
- The OASIS becomes a new home for the development of the MQTT
- The recent version of MQTT is 5.0, which is a successor of the 3.1.1 version. Version 5.0 is not backward, comfortable like version 3.1.1

Characteristics

- It is a machine to machine protocol, i.e., it provides communication between the devices
- It is designed as a simple and lightweight messaging protocol that uses a publish/subscribe system to exchange the information between the client and the server
- It does not require that both the client and the server establish a connection at the same time
- It provides faster data transmission, like how WhatsApp/messenger provides a faster delivery. It's a real-time messaging protocol.
- It allows the clients to subscribe to the narrow selection of topics so that they can receive the information they are looking for

Architecture

- The architecture includes
    - Message
    - Client
    - Broker
    - Topic

Message

- The message is the data that is carried out by the protocol across the network for the application
- When the message is transmitted over the network, then the message contains the following parameters:
  - Payload data
  - Quality of Service (QoS)
  - Collection of Properties
  - Topic Name

Client

- In MQTT, the subscriber and publisher are the two roles of a client
- If any program or device uses an MQTT, then that device is referred to as a client
- A device is a client if it opens the network connection to the server, publishes messages that other clients want to see, subscribes to the messages that it is interested in receiving, unsubscribes to the messages that it is not interested in receiving, and closes the network connection to the server
- In MQTT, the client performs two operations:
  - **Publish**
    - When the client sends the data to the server, then we call this operation as a publish
  - **Subscribe**
    - When the client receives the data from the server, then we call this operation a subscription

Server

- The device or a program that allows the client to publish the messages and subscribe to the messages
- A server accepts the network connection from the client, accepts the messages from the client, processes the subscribe and unsubscribe requests, forwards the application messages to the client, and closes the network connection from the client

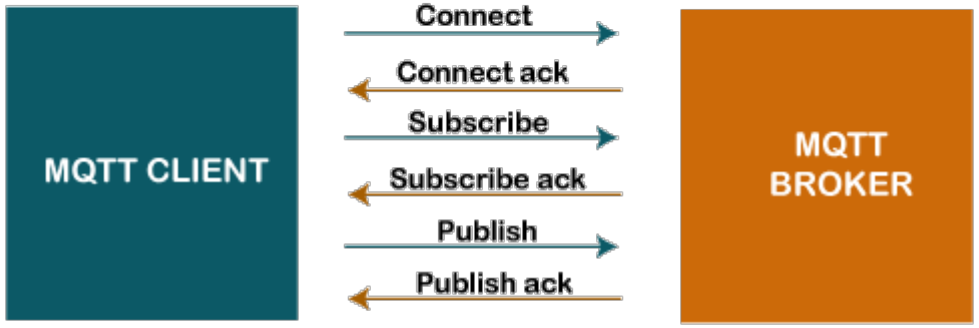
Topic

- The label provided to the message is checked against the subscription known by the server is known as TOPIC
- It is a collection of messages
- A publisher can publish a message to a topic or a subscriber can subscribe to a topic to receive new messages posted

Message Format

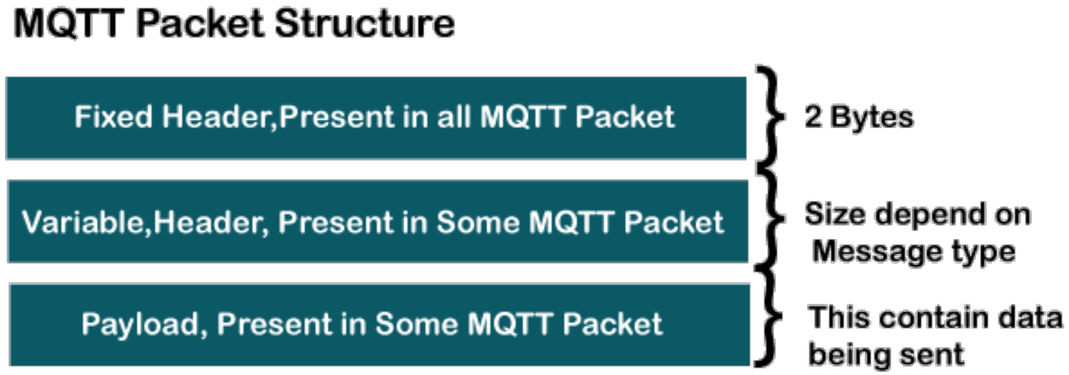
- The MQTT uses the command and the command acknowledgment format, which means that each command has an associated acknowledgment
- The connect command has connect acknowledgment, subscribe command has subscribe acknowledgment, and publish command has publish acknowledgment
- This mechanism is similar to the handshaking mechanism as in TCP protocol

MQTT Message Format



Packet Structure

- The MQTT message format consists of 2 bytes fixed header, which is present in all the MQTT packets
- The second field is a variable header, which is not always present
- The third field is a payload, which is also not always present
- The payload field basically contains the data which is being sent
- Some commands do not use the payload field, for example, disconnect message



MQTT Packet Structure

Fixed Header,Present in all MQTT Packet

2 Bytes

Variable,Header, Present in Some MQTT Packet

Size depend on Message type

Payload, Present in Some MQTT Packet

This contain data being sent