Internet of Things: Constrained Application Protocol

Pallavi Vyas, Navpreet Kaur, Rishi Chopra
School Computer Science and Engineering,
Lovely Professional University, Phagwara, Punjab.

Abstract

Machines are the tools that give the human a leverage to do a task in less time and efficiently. Every one now a day is surrounded with lot of machines but a limitation of machines is that we need a person to operate them these machines cannot run without a human .M2M (Machines to Machine communication) is the solution of that problem . Internet of Things (IoT) is subset of M2M and it takes the machines to machine communication to another level. Internet of Things makes use of internet to communicate with the things. Here thing can be any machine that have a unique identity on internet. Internet of Things allows machines to communicate with other machine as well as control them .This paper describe the basic structure of Internet of Things and how things work in Internet of Things. We describe CoAP (Constrained Application Protocol) that is the backbone of Internet of Things.

Introduction

The internet of things is a broad concept of connecting all the electrical and the embedded devices to the internet and making them able to communicate with each other. The word "thing" in the internet of things represents a super set of all the natural and man-made entities for example a man or an animal with a biochip or any vehicle with the inbuilt sensor to notify the driver when the air pressure of the tire is low.

The IOT concept is to connect all such entities by the internet by which they can send and receive the data over the web without any interference of a human. The advantage of such type of concept inn real world implementation can result in many ways like in fully automated house, in medical field by providing the alerts related to the critical health issues. This can be also very much helpful to generate the system for serving as an assistant for the physically challenged peoples.

There is a variety and a wide area full of with the application of the IOT these can be monitoring the efficiently use of electricity, monitoring your garden when we are not at our home, maintaining the city clean and pollution free by tracking their increasing rate in certain area of the city and making necessary effort according to that.

"Internet of Things" term is introduced by Kevin Ashton in 1999. This is extension to M2M communication and takes the M2M communication beyond the limits. IoT introduce the new environment where all the embedded devices, use the new set of protocols to connect and interoperate other devices.

Application areas of IoT

IoT is such a broad topic that covers a large number of applications in every area. Some important and common of them are given below:

1. Building Automation

Every building and home includes lots and lots of mechanical and electrical systems. IoT can be helpful to monitoring and controlling these systems. We are able to control the light, ventilation, air conditioning,

heating, security etc. of a building and utilize the energy in better and efficient manner. A big part of the greenhouse gas is produced by the building. IoT helps us in building the smart building that can maintain the use of natural sources and reduce the energy consumption.

2. Transport systems

IoT can solve the large number of problems of transport system. A small traffic sensor can identify Traffic jams and send the information of traffic jam to other vehicles on the same road so they can reroute. Toll system can be improved. Vehicle can communicate with other vehicle on the road. Smart vehicle monitoring system sends the driver various real time data of the car like tires pressure, suggested car speed. Vehicle can also monitor the behavior of the driver and send the suggestion of them by comparing the driving behavior of other vehicle on the same road.

3. Medical systems

Health care can become easier by IoT. Doctors can monitor their patients every time. Sensors can be attached with the patients and these sensors produce the real time data related to the body functions like blood pressure, heart rate, temperature etc. Produced data is send to the doctor and a system keep the monitor on the data and in case of any emergency system notify to the doctor and send ambulance to the patient home.

4. Smart cities

IoT can make the smart city possible where environment is monitor machines different sensors monitor the environment and produce the data and other machines use that data. For example, a temperature sensor tells the air conditioning system the current temperature and air conditioning system takes the action accordingly.

5. Retail, logistics and supply chain management

Suppose new stock is received by a warehouse .in this case we have to update the database manually. RFID can be used to identify the new stock and it can also be useful to update the data of database without any person.

Technologies

1. RFID

Radio-frequency identification make use of electromagnetic fields to transfer data .RFID have some information stored inside and the information is transferred using radio waves or electromagnetic. Main use of them is to identify items.

2. Sensor

To monitor different things we need some sensors that are capable of identify the change in state of objects. Sensors monitor the object and produce the data and send that data to the computing devices like data centers and embedded microcontrollers.

3. Microcontrollers

Small and embedded devices need processing unit that can help in taking decisions. Microcontrollers are the computer chips that provide the ability to compute for small devices and embedded systems.

4. Protocols

Protocols are rules that make the communication possible between the objects in IoT .Internet protocol (IP) is standard for making things online so IoT also utilize the IP .IoT needs more number of IP address to make each and every thing online so it uses IPv6.some other protocols used in the IoT are Constrained Application Protocol (CoAP), ZigBee, HART etc.[2]

Basic Protocols used in IoT

➢ **CoAP (Constrained Application Protocol)**

The CoAP protocol is designed by the CORE (Constrained RESTful environments) group, a sub group of ITEF(internet engineering task force). CoAP was specifically designed for the purpose of the communication of the different constrained nodes via constrained network.[2]

1. RESTful (representational state) environment.
2. Constrained Nodes - These are the small devices with limited RAM and ROM, power and processing power (for e.g. sensors).
3. Constrained Networks - A network with lower transfer rates and a limited stack capabilities and packet delivery probability. Example: LoWPAN(low power wireless personal area network).

The main objective of designing the CoAP protocol is to provide a general protocol for web in order to fulfill the requirements of constrained devices to communicate.

The CoAP protocol is an optimized form of http protocol for M2M communication, with inbuilt features like asynchronous message exchanges, multicast etc. Some advanced features provided by CoAP are given below:

1. It uses UDP (user datagram protocol) for asynchronous messaging.
2. It supports low overhead message format.
3. It provides content type support.

Along with these feature, CoAP support the request/response (client/server) model, which proves its similarity with the http protocol.

CoAP message types

There are specifically 7 types of messages used in CoAP protocol defined by IETF.

1. **Confirmable Message (CON)** - A request made by this type of message requires a message in response. The response message may be either an acknowledgement message or a reset message. There may be a case when response is not received than the request is retransmitted again and again until request the available attempts.
2. **Non-confirmable Message (NON)** - A request made by NON message does not require any response.
3. **Acknowledgement Message (ACK)** - ACK message is send in response to give the information to the client that the CON message arrived. It only indicates the successful arrival of CON message but not the failure or success request.
4. **Reset Message (RST)** - When there is any kind of fault (missing context) in receiving the CON/NON message then the RST message is send.
5. **Piggy-backed Response**- When the request is successfully processed than it sends the ACK message which also contains the response data. This response data is called as the piggy backed response.

6. **Separate message** - In case when the server requires some time I order to process the request greater than the acknowledgement (with empty message)wait time (ACK_TIMEOUT), so to avoid the retransmission of the request by client ,the server immediately transmit an ACK message without the response data and sends the piggy backed response with a separate message. The response data is send by separate message in case of NON message because no ACK is send in case of NON request.
7. **Empty Message** -This message contains a response code 0.00, containing only 4 bytes header. We will discuss about response codes later in this paper.

Negotiation of content

In CoAP the content negotiation is supported which makes it similar to http. For this purpose the client use an option provided in CoAP i.e. "accept " to tell the server what format it is looking for and the server when replies, it use the option "Content-Type" to inform the client what they are going to receive.

Security

As by now you know that the CoAP use UDP not TCP so SSL/TLS are not available to use. So DTLS (Datagram Transport Layer Security) is used for connection Encryption.

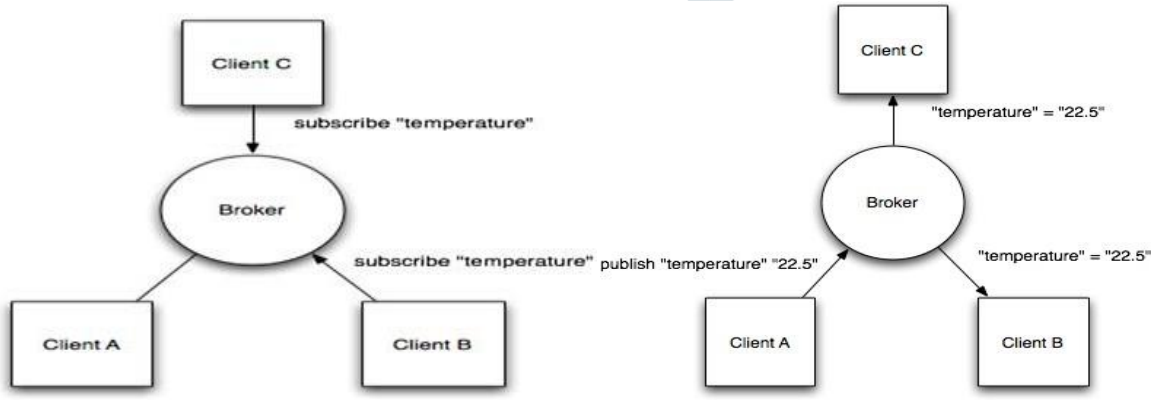
- **MQTT (Message Queue Telemetry Transport) protocol**

The invention of MQTT protocol was done by IBM and EUROTECH in 1999. It was donated to eclipse project eclipse paho. It is a lightweight machine to machine (M2M) messaging protocol.

Architecture of MQTT

The architecture of MQTT protocol consists of single application layer and supports publish/subscribe topic based approach for communication.

In its architecture sensors are clients which can publish/subscribe messages to each other but via central server called as "BROKER". The TCP/IP protocol is used for messaging in MQTT. It means every message must go to BROKER and then the BROKER decides where to route that message. Messages are opaque to the BROKER. Let's take an example scenario having 3 clients and a broker in a network. As you can see in fig. below that the client B and C are subscribed to topic temperature via Broker.



After some time client A publish a message (value) to topic temperature, than Broker immediately forwards the message to all subscribed clients.

PUBLISH/ SUBSCRIBE Topic Based Approach

In this approach the message published by the client(sensor) is published to an address which is known as a TOPIC. Now the other clients can subscribe to that topic and get all the messages published to that topic. A client can subscribe to multi number of topics. Hence by this approach MQTT clients can communicate in one-to-one, one-to-many or many-to-one manner.

Security: for security purposes MQTT Broker requires a username and password for authentication. The SSL/TLS is also used for connection encryption.

Types of messages (Quality of service): there are three messages supported in this protocol, those are Fire and Forget, Delivered at least once and delivered exactly once.

Comparison between HTTP and CoAP

CoAP and http both are web oriented protocols. CoAP involves some of the features of http but not completely same as http. CoAP adopt the features like abstraction of resources, RESTful interactions, and the header options extensibility. These features allow CoAP to fetch the simple documents from application platforms.[2]

HTTP allows us to combine different sources or services with very little scripting effort called “mash up”. Http allows only one-to-one communication, whereas the CoAP allows the independent communication among RESTful methods.

HTTP uses the TCP for communication which is very heavy for constrained devices but the COAP uses the UDP which is lighter than TCP[1]. Group communication is a significant requirement for the Internet of Things, for instance for automation applications. To make up for the unreliability of UDP, CoAP defines transactions with retransmissions[1].

REFERENCES

1. Constrained Application Protocol (CoAP) draft-ietf-core-coap-18 [https://tools.ietf.org/html/draft-ietf-core-coap-18]

2. https://www.researchgate.net/publication/229057545_Implementation_of_CoAP_and_its_Application_in_Transport_Logistics