Volume 8, No. 3, March – April 2017
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
REVIEW ARTICLE
Available Online at www.ijarcs.info

A Survey on Application Layer Protocols for Internet of Things (IoT)

Makkad Asim
Department of Computer Science and Engineering
Institute of Technology, Nirma University
Ahmedabad, India - 382481

***Abstract*:** Internet of Things (IoT) or Web of Things (WoT) is emerging technology and it wireless network between two or more smart objects or smart things connect via Internet. IoT classified in two type first is inside of IoT and second side is outside of IoT. In inside of IoT consider as protocols in IoT. In outside of IoT consider as sensor, actuators, etc..., those are physically possible. In inside of IoT consider as Protocols and IoT have their own protocol stack. Protocol stack have different layer like Application layer, Transport layer, Internet layer and Physical/Link layer.

The judgmental role goal of IoT is to ensure effectual communication between two objects and build a sustained bond among them using different application. The application layer responsible for providing services and determining a set of protocol for message passing at the application layer. This survey understand application layer protocol like CoAP, MQTT, AMQT, XMPP and RESTFUL. Also describe some of the new protocols in application layer protocol.Which type of architecture (like request/response, client/server and publish/subscribe) and security (like DTLS, TCL/SSL and HTTPS) support in those protocols.

Keyword: Internet of Things (IoT), Application layer protocols, CoAP, MQTT, AMQT, RESTFUL, Web-socket.

I. INTRODUCTION

Internet of Things is environment where small smart devices are connected always, anytime and anywhere with each other via internet. A question is every one mind why we use IoT?, because of IoT enable all kinds of devices to connect together and share information seamlessly and number of things connected to the Internet is more than people present on earth that reason we used IoT. Those things or objects is small embedded devices that must contain low power and low cost also. Generally IoT classified in two type first is outside of IoT and second is inside of IoT. In outside of IoT types physically possible, outside of IoT are RFID (Radio Frequency Identification), WSN (Wireless Sensor Networks (for example Sensors and actuators)), Addressing schema, DSA (Data Storage and Analysis) and Visualization[2]. Those five element enable IoT component enable physically.

Another side of IoT is outside of IoT, in this type include protocols. IoT have its own protocol stack, their different than other protocol stack like OSI model and TCP/IP protocol stack. IoT model protocol Stack show in figure below. Figure show different layer of model layer like Application layer (Protocols are COAP, MQTT, AMQP, XMPP, RESTFUL and Web-sockets), Transport layer (Protocols are UDP and DTLS), Internet layer (Protocols are RPL and 6LoWPAN) and Physical/Link layer (protocols are IEEE 802.11 series and IEEE 802.15 series) [7]. In this survey paper we talk about only application layer, In general manner IoT application layer protocols just replace TCP/IP application layer protocols in IoT framework.

Definitions - The Internet of Things (IoT) is the inter-networking of physical devices, vehicles, buildings, and other items-embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data[1]. In short IoT connect with each and every object and communicate with each other anywhere and anytime, so IoT do just like Machine to Machine (M2M) communication. In other word Internet that make things or Objects are smart call IoT.

Layer	Protocols
Application Layer	CoAP, MQTT, XMPP, AMQP, RESTFUL, Websockets
Transport Layer	UDP, DTLS
Internet Layer	RPL, 6LoWPAN
Physical/Link Layer	IEEE 802.15 Series, IEEE 802 11 series

Figure 1: IoT protocols stack

II. RELATED WORK

Internet of Things (IoT) is emerging technology. Show in previous section inside of IoT describe as protocols. By studying paper regarding IoT and IoT protocol related IETF standards paper show application layer protocols focus basically on message exchange between applications and the internet [2]. Most of paper summarize some the most important standard that are provide different stander organizer. It also provides a discussion of different IoT challenges including mobility, scalability. In other survey paper show different layer like transport layer used or provide security in application layer protocols. Internet layer protocols like RPL (Routing for low power and lossy network) and 6LoWPAN (IPv6 over Low Personal Area Network). 6LoWPAN used in

application layer as providing IP address to devices for communication. Physical layer protocols like IEEE 802.11 series,IEEE 802.15 series, zigbee and Adriano. Those physical layer protocols used at application layer to manage sensor and actuators. Application layer work with other layer like transport layer, Internet layer and physical layer.In this paper our aim to provide comprehensive survey to describe all main six application layer protocols and also provide newly arising standards protocols and their architecture.

III. APPLICATION LAYER PROTOCOLS

This section reviews standards and protocols for message passing in IoT application layer proposed by different standardization[2]. All-most web-based application and IoT application are IP based and they use TCP and UDP for transport. However, there are several message distribution functions that are common among many IoT applications; it is desirable that these functions be implemented in an interoperable standard ways by different applications. Those protocols are:

A. MQTT

MQTT (Message Queue Telemetry Transport) was develop by or introduce by IBM in 1999 and standardized by OASIS in 2013 to target come up with lightweight M2M communication[3]. It is publish/subscribe protocol architecture similar to client/server protocol show in figure below. The importance of MQTT protocol is due to its simplicity and the no need of high CPU and memory usage (reason is the lightweight protocol) [6]. MQTT supports a wide range of different devices and mobile platforms. At transport layer TLS/SSL security provide to MQTT.

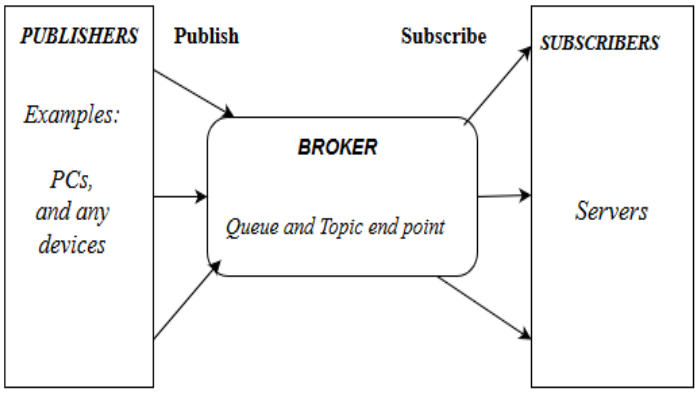

Figure 2: MQTT Architecture

Show above figure there three component are there publishers, a broker and subscribers. Publishers are generally lightweight sensors that connect with a broker and send data to a broker and go back to sleep. Subscribers are IoT applications that interested in data send by sensors and also connect with a broker, so broker send interested data to subscribers. The brokers classify sensory data in topics and send them to subscribers interested in the topics. This all thing is on IoT point of view.

MQTT provide 3 option to achieve message in Quality of Services (QoS):

1. *One delivery (at most):*

Deliver message according to best try of the network. An acknowledgment is not required. Lowest level of QoS.

2. *One delivery (at least):*

At least one message can be send and some duplicate message are there. An acknowledgment is required

3. *On delivery:*

Additional protocol required to ensure that one and only one message send. It is highest level of QoS.

B. AMQP

The Advanced Message Queuing Protocol (AMQP) is a protocol that across from the financial industry. Security is manage with the use of the TLS/SSL protocols. Its run over TCP. AMQT is follow publish/subscribe communication protocol for messaging [6]. AMQP is same like MQTT but AMQT have advantage its store data then forward it, and this feature used at when network disruptions that time ensures reliability. Show in figure below a broker divide into two part exchange and queue. Exchange responsibility to receive publishers messages and distribute to queue. Queues is based on pre-define roles and condition and it's basically send message to subscribers who subscribe those data.

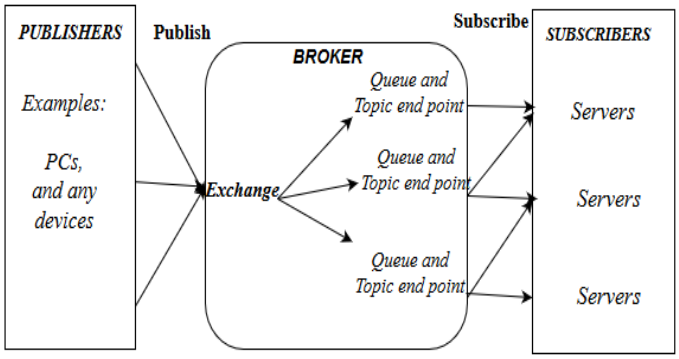

Figure 3: AMQP Architecture

C. CoAP

CoAP (constrained application protocol) is used for low power and low memory embedded devices where it can be used for communication instead of HTTP. Currently there is HTTP protocol available with request/response paradigm but HTTP has many features and more footprint [5]. HTTP runs over TCP where TCP will need more resources due to three way handshake and many more complex mechanism. Now for low power embedded devices, there is no need of this heavy protocols and we can optimize it to run over TCP.

As CoAP is a Restful web transfer protocol for use with constrained networks. CoAP uses client/server model of approach same as HTTP. It is designed for constrained

networks with low overhead and lower footprint. Some points for CoAP that makes better protocol compared to HTTP is:

- CoAP runs over UDP (User data-gram protocol) that helps to avoid costly TCP handshake before data transmission
- CoAP protocol is only 4-byte header and provides reliable transfer and no reliable transfer as it uses four types of messages.
- Show above figure, its support four types of message 1) Confirmable, 2) Non-Confirmable, 3) Acknowledgement and 4) Reset. Request/Response layer used those message and classified in 1) Piggy-backed, 2) Separate response, 3) Non confirmable request and response, and communicate with each other show in below figure.

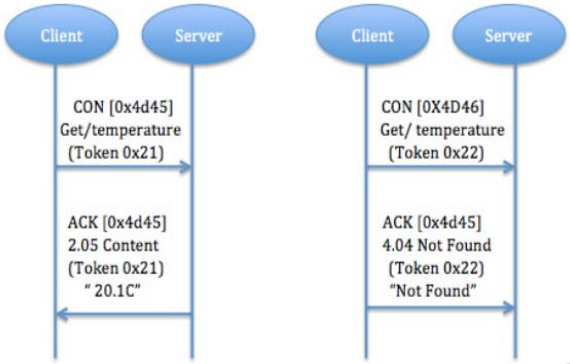

Figure 4: In piggy-backed, successful and failure response results of GET method [3].

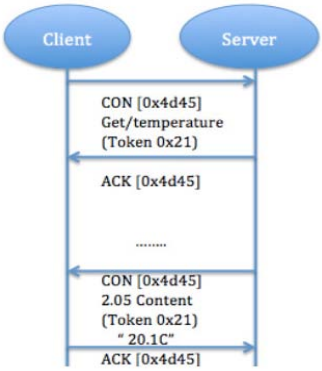

Figure 5: A Get request with a separate response [3].

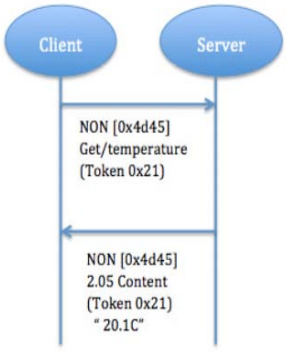

Figure 6: Non confirmable request and response [3].

As in HTTP, CoAP utilizes GET, PUT, PUSH, DELETE messages requests to retrieve, create, update, and delete, respectively [5].

D. RESTFUL SERVICES

Representational State Transfer (RESTFUL Services) is an engineering that gives web administrations which permit correspondence and information trade between various gadgets utilizing HTTP in IoT condition [5]. REST utilizes the HTTP strategies GET, POST, PUT, and DELETE to give an asset arranged informing framework where all activities can be performed essentially by utilizing the synchronous request/response HTTP commands.

RESTful services use the secure and reliable HTTP which is the proven worldwide Internet language. It can make use of TLS/SSL for security.

E. WEB-SOCKET

The Web-Socket protocol provides two ways for communication between clients and a remote server. Web-Socket provides security similar to the security model used HTTPS protocol. For browsing application layer used and web-socket work on TCP transport layer protocol, so they need to interact and communicate with host those who connect with remote. Web-Socket is a web-based protocol that works on the single TCP channel and provides full duplex communications. Web-socket starts session without publish/subscribe and request/response models like previous protocols [6].

F. XMPP

Extensible Messaging and Presence Protocol (XMPP) is a messaging protocol that was designed originally for chatting and message exchange applications. It was standardized by IETF more than a decade ago. In all application layer protocols only XMPP protocol support publish/subscribe and request/response model and it's depend on application developers to develop application which model they use [7]. It does not provide any quality of service guarantees and, hence, is not practical for M2M communications. XMPP is rarely used in IoT but has gained some interest for enhancing its architecture in order to support IoT applications.

E. DDS

Data Distribution Service (DDS) is another publish/subscribe protocol that is designed by the Object Management Group (OMG) for M2M communications. It defines two sub layers: data-centric publish- subscribe and data-local reconstruction sub layers [8].

The first takes the responsibility of message delivery to the subscribers while the second is optional and allows a simple integration of DDS in the application layer. Publisher layer is responsible for sensory data distribution. Data writer

interacts with the publishers to agree about the data and changes to be sent to the subscribers. Subscribers are the receivers of sensory data to be delivered to the IoT application. Data readers basically read the published data and deliver it to the subscribers and the topics are basically the data that are being published. In others words, data writers and data reader take the responsibilities of the broker in the broker-based architectures.

G. SMQTT

An extension of MQTT is Secure MQTT (SMQTT) which uses encryption based on lightweight attribute based encryption. The main advantage of using such encryption is the broadcast encryption feature, in which one message is encrypted and delivered to multiple other nodes, which is quite common in IoT applications [8]. In MQTT architecture exchange data between publishers and subscribers SMQTT use encryption algorithm them. DDS and SMQTT both protocols are new emerging protocols, and both are similar with MQTT in show both are upgrades of MQTT.

IV. CHALLENGES

A. SEQURITY

Sensors will sense the data and send it to network so confidentiality of information is very important where we can use some standard encryption/decryption to encrypt the data and send over network and other device can decrypt it. That's why application layer security.

B. RELIABILITY

The main challenge in IoT is reliability. When one IoT node send data to more than one server, if one of the server will crash or goes down then it is very hard to get original file. If file will be deleted at server side it cannot be reconstructed so, data will be lost.

C. LOW POWER

The one of the main challenge for IoT is low power of devices as embedded devices used in IoTs are deployed at many places and that has limited power capacity in this case it's very important to save power whenever its possible [3]. So there is need of mechanism where we can power off the devices when there is no need of power and can power up again whenever needed.

D. NETWORK CAPABILITY

The challenge regarding network capability is there are many sensors and devices connected with network and the data from sensor device will be sent through wired or wireless interface. The transmission system or network should be able to collect all the data from sensors and make sure that no data loss occur due to network congestion.

V. CONCLUSION & FUTURE WORK

There are three major components for implementing IoT on different applications: Security, Privacy and Trust. While increasing the growth of IoT, Security is more important for reliable data transferred among the billions of smart objects. In these research i concentrate on CoAP protocol application layer protocol on IoT devices. Having light weight and consume low energy, CoAP is used on many applications of IoT. Here describe all protocols in IoT application layer protocol. Summary of those protocol given in below figure.

To secure data transferred, CoAP combined with DTLS protocol named as Datagram Transport Layer Security protocol as the security agent. So in future we focus in Security in Application layer protocols.

Protocols	Transport	QoS	Architecture	Security
CoAP	UDP	YES	Request/Response	DTLS
MQTT	TCP	YES	Publish/Subscribe	TLS/SSL
XMPP	TCP	NO	Request/Response Publish/Subscribe	TLS/SSL
RESTFUL	HTTP	NO	Request/Response	HTTPS
AMQP	TCP	YES	Publish/Subscribe	TLS/SSL
Web socket	TCP	NO	Client/Server Publish/Subscribe	TLS/SSL
DDS	TCP/UDP	YES	Publish/Subscribe	TLS/SSL
SMQTT	TCP	YES	Publish/Subscribe	It have own

Figure 7: Summary of application layer protocols

VI. REFERENCES

[1] Administration. Internet of Things. 2016 (accessed November 3, 2016). URL: https://en.wikipedia.org/wiki/Internetofthings.

[2] Jayavardhana Gubbi et al. "Internet of Things (IoT): A vision, architectural elements, and future directions". In: Future Generation Computer Systems 29.7 (2013), pp. 1645–1660. ISSN: 0167-739X. DOI: http://dx.doi.org/10.1016/j.future.2013.01.010. URL: http://www.sciencedirect.com/science/article/pii/S0167739X13000241.

[3] Xi Chen. "Constrained Application Protocol for Internet of Things". URL: https://www.cse.wustl.edu/~jain/cse574-14/ftp/coap/.

[4] S. Kraijak and P. Tuwanut. "A survey on IoT architectures, protocols, applications, security, privacy, real-world implementation and future trends". In: 11th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2015). Sept. 2015, pp. 1–6. DOI: 10.1049/cp.2015.0714.

[5] IoT Messaging Protocols. 31 march 2015.URL: https://iotprotocols.wordpress.com/.

[6] Sangyoon Oh, Jai-Hoon Kim, and Geoffrey Fox. "Real-time Performance Analysis for Publish/Subscribe Systems". In: Future Gener. Comput. Syst.26.3 (Mar. 2010), pp. 318–323.

ISSN: 0167-739X. DOI: 10.1016/j.future.2009.09.001. URL: http://dx.doi.org/10.1016/j.future.2009.09.001.

[7] Stan Schneider.Understanding the Protocols Behind TheInternet Of Things.URL: http://electronicdesign.com/iot/understanding-protocols-behind-internet-things

[8] What is MQTT? URL: http://internetofthingsagenda.techtarget.com/definition/MQTT-MQ-Telemetry-Transport