Data transfer protocols in IoT-an overview

Article *in* International Journal of Pure and Applied Mathematics · January 2018

CITATION
1

READS
262

2 authors:

Tanya Mohan Tukade
University of Texas at Dallas
1 PUBLICATION **1** CITATION
SEE PROFILE

Rajeshwari Banakar
B.V. Bhoomaraddi College of Engineering and Technology (BVBCET)
55 PUBLICATIONS **951** CITATIONS
SEE PROFILE

Data Transfer Protocols in IoT-An Overview

Tanya Mohan Tukade, R M Banakar
School of Electronics and Communication Engineering
B.V. Bhoomaraddi College of Engineering and Technology, Hubballi, India
Email: tanyatukade.123@gmail.com, banakar@bvb.edu

***Abstract*—This paper provides an overview of the Internet of Things (IoT) with emphasis on network communications and messaging protocols in IoT. IoT is expected to bridge diverse technologies to enable new applications by connecting physical objects together in support of intelligent decision making. The paper elaborates the relationship between machine to machine communication (M2M) and IoT. It also highlights different data transfer messaging protocols namely COAP, MQTT, XMPP, AMQP and HTTP. It further discusses the specialities and properties of these protocols. Lastly it addresses the latest technology-LoRA (LoRAWAN Protocol) and its smart applications in real time.**

***Index Terms*—IoT, MQTT, COAP, HTTP, AMQP, XMPP, LoRA, Data Transfer protocols, M2M.**

I. INTRODUCTION

IoT has significantly changed ones perspective of living style. It has enabled many non-living objects to behave smartly and intellectually according to the circumstance and environment. A growing number of physical objects are being connected to the Internet at an unprecedented rate realizing the idea of the Internet of Things (IoT). The IoT envisions hundreds or thousands of end-devices with sensing, actuating, processing, and communication capabilities able to be connected to the Internet [1]. IoT techniques are rapidly getting into the field of automation, agriculture, medicine, Transportation and technology. By 2020 it is expected that 50 billion devices will be connected through IoT. In this rapidly changing IoT technology new devices are being designed with this technique enabled. Over time, the IoT is expected to have significant home and business applications, to contribute to the quality of life and to grow the worlds economy. Communicating relevant physical parameter data and transmitting this data using internet is the buzz activity in todays state of art technology world.

IoT design platform consists of the distinguishing choice of communication media as its first goal. The main Hardware design are the smart objects namely, sensors and actuators. These devices can be directly connected using cellular technologies such as 2G/3G/Long Term Evolution and beyond (5G) or they can be connected through a gateway, forming a local area network, to get connection to the Internet. The smart objects are application specific and are customised to the design scenario. These sensors are the source of data acquisitions in the IoT design platform. Based on the selection of the sensor and the action to be performed the actuators are chosen. This forms the basic hardware entity unit. [12] The captured data needs a direction to be transferred, modified, controlled, acknowledged, stored or exported to other devices. These tasks can be performed through suitable protocols. The smart objects are meaningless and lifeless without the associated data acquisition software process. The smart objects should be able to communicate amongst themselves. This can be termed as internode communication. A suitable wireless network should be setup to make appropriate and knowledgeable communication.

The data transfer should be using appropriate processing units. These processing units should be programmable and reliable data portability is the main criteria. Choice of suitable development boards depends on the design scenario and cost effectiveness.The source-destination communication should be properly established. It depends on the application design platform. There should be proper compatibility between the source hardware/software module and the user service module. In IoT one should clearly plan the need for near field communication and Far field communication. The integration of the different IoT protocols to deliver desired functionalities is the need of the hour.

In section 2 introduction to the data transfer protocol between the transmitter and the receiver is given. In section 3 brief discussion of the himan to machine communication involved in the context of messaging protocols is presented. Section 3 also deals with the salient features of M2M communication. The messaging protocols are given in section 4. Section 5 presents the long range data communication media namlely LoRA. Section 6 gives the conclusion of the work, where the comparision of the messaging protocols is important to choose a suitable design platform.

A. History of protocols

A data transfer protocol is a standardised format for transmitting data between two devices. The type of protocol used can determine the variables. The FTP protocol, enabling file transfers between remote systems, was first published as a “Request for Comments” (a collection of technical and organizational notes about the Internet) on April 16, 1971. Since its inception, FTP has been the standard protocol used to transfer files between remote computers. The User

Datagram Protocol (UDP) is one of the core members of the Internet protocol suite. The protocol was designed by David P. Reed in 1980 and formally defined in RFC 768. With UDP, computer applications can send messages. The most popular network protocol in the world, TCP/IP protocol suite, was designed in 1970s by 2 DARPA scientists Vinton Cerf and Bob Kahn, persons most often called the *fathers of the Internet*.

The Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, hypermedia information systems. Development of HTTP was initiated by Tim Berners-Lee at CERN in 1989. Netscape Communication created HTTPS in 1994 for its Netscape Navigator web browser. Originally, HTTPS was used with the SSL protocol. As SSL evolved into Transport Layer Security (TLS), the current version of HTTPS was formally specified by RFC 2818 in May 2000. MQTT was invented by Andy Stanford-Clark (IBM) and Arlen Nipper (Arcom, now Cirrus Link) back in 1999, when their use case was to create a protocol for minimal battery loss and minimal bandwidth connecting oil pipelines over satellite connection.

AMQP was originated in 2003 by John O'Hara at JPMorgan Chase in London, UK. From the beginning AMQP was conceived as a co-operative open effort. Initial development was by JPMorgan Chase from mid-2004 to mid-2006 who contracted iMatix Corporation for a C broker and protocol documentation. XMPP was developed by Jabber. The protocol was developed by the Jabber open-source community in 1999 for near real-time instant messaging (IM), presence information, and contact list maintenance. Designed to be extensible, the protocol has been used also for publish-subscribe systems, signalling for VoIP, video, file transfer, gaming, the Internet of Things (IoT) applications such as the smart grid, and social networking services. Typically, IoT is expected to offer advanced connectivity of devices, systems, and services that goes beyond machine-to-machine (M2M) communications and covers a variety of protocols, domains, and applications.

The journey from 1969 to 2017 in computer networks from the inception of ARPANET in 1969 to the present wireless sensor networks is interesting. Growth in the features of data transfer protocol suite during these 48 years is progressing in a better direction. This is clearly visible in the projection of connectivity of 50 billion devices by the year 2020. The future, at least in the next five years depends on how best the device integration, network integration, optimal sensor and actuator connectivity takes its shape. The amalgamation of IoT suite and wireless sensor network is seeing the light of the day in various application scenarios. In particular to address the data transfer protocols suite needs special attention. These data transfer protocols can be categorized into two types namely;
1) File transfer protocols
2) Messaging protocols
File transfer protocols are suited for web applications whereas messaging protocols are best suited for IoT framework. The file transfer protocols are not suited in the existing format for IoT, since IoT nodes are basic sensor nodes used in raw data collection from the application scenario of framework.

TABLE I: Evolution of Protocols

Protocols	Year
FTP	1971
UDP	1980
TCP/IP	1983
HTTP	1989
HTTPS	1994
COAP	1997
MQTT	1999
XMPP	1999
AMQP	2003
LORA [LoRaWAN Protocol]	2009

File transfer protocols are suited for web applications whereas messaging protocols are best suited for IoT framework. The file transfer protocols are not suited in the existing format for IoT, since IoT nodes are basic sensor nodes used in raw data collection from the application scenario of framework.
There are two types of prominent data command and transfer that may occur. One is th Human to Machine interface. The other is Machine to Machine interface.

II. HUMAN TO MACHINE COMMUNICATIOS IN IoT [HMI]

Human to machine communication originally emerged from telemetry technology, and its main aim was to measure data and automatically transmit it from remote sources typically by cable or a radio. Nowadays, plethora of sensors are being developed, which have better perceptual abilities than humans and can detect information that humans cannot. Affordable electronic devices have led to an increasing number of them being connected to the Internet. The smart IoT devices open up the possibility to reduce the burden on the user end by equipping everyday objects, such as a wheelchair, with decision-making capabilities. Sensors have been used in communication using HMIs for years. Human to Machine communication is a very important development in Internet of Things. Even though the fundamental concept of collecting and sharing data through internet stays same with H2M communication, the source from

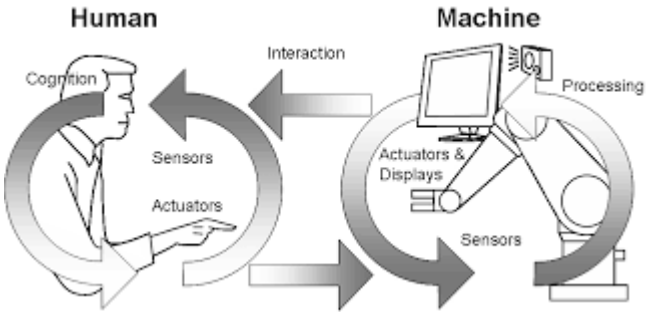

Fig. 1: Human to machine interaction

which the data is collected will make the difference. A very simple example would be someone with a chip implanted to monitor the heart rate. This chip will collect the information automatically and will alert the doctors on their smart phones when heart rate is abnormal.

H2M communication has revolutionized the personal health sector, which also has a direct impact on insurance industry. There is almost 200 percent growth in fitness wearables market in 2015. The greater the role of the HMI, the more important it is to select one with a high degree of scalability. It is important that one provides proper operation control and connectivity to supervisory systems. In motion applications, HMIs must also be able to respond quickly to commands as well as unanticipated situations requiring complex manoeuvres. Fig 1. shows the human to machine interaction and its relation to the IoT. Fig 1. shows the elaborate human to machine interaction which demonstrates the use of sensors, actuators, cognition unit and processing unit.

The data transfer in HMI model is based on the cognitive ability of the human. Suppose on a wheel chair the person wants to move right, this is the cognitive message which helps the human action to press the right move direction button. This enables the machine interface to recognise the input and the wheels move towards right with the help of the actuators. The machine sends the control data to the actuator unit to perform the required action. In some design unit the human interface may be through voice processing. Example, with a simple command “right”.

III. MACHINE TO MACHINE COMMUNICATIONS IN IoT [M2M]

Machine to machine (M2M) is a broad term that can be used to describe any technology that enables networked devices to exchange information and perform actions without the manual assistance of humans. M2M communication is often used for remote monitoring. M2M communication is an important aspect of warehouse management, remote control, robotics, traffic control, logistic services, supply chain management, fleet management and telemedicine, transportation. It forms the basis for a concept known as the Internet of Things (IoT). Key components of an M2M system include sensors, RFID, a Wi-Fi or cellular communications link and autonomic computing software programmed to help a networked device interpret data and make decisions. The most well-known type of M2M communication is telemetry. Currently, M2M does not have a standardized connected device platform and many M2M systems are built to be task- or device-specific. M2M communications expands telemetry’s role beyond its common use in science and engineering and places it in an everyday setting. People already are using M2M, but there are many more potential applications as wireless sensors, networks and computers improve, benefitting the concept to be amalgamated with other technology.

A point worth stressing is that data transfer patterns in the M2M-driven Internet of Things will differ fundamentally from those in the classic ‘human-to-human’ (H2H) internet. M2M finds its applications in Smart homes, healthcare, infrastructure, automotive and transport, supply chain, Retail, field service, utilities-smart metrics and grids, security and surveillance, environmental monitoring, agriculture and military. M2M is certainly happening, but the market is fragmented into numerous verticals. Right now there are around 110 million M2M devices connected to the internet. This year it is expected to climb to 400 million, and expects this to grow to 18 billion by 2022. Many of these devices will be used to link the physical world to the internet via sensors that take readings from their local environment and output the information up into the cloud [14]. M2M devices are usually small and inexpensive, introducing energy, bandwidth, computation, and storage constraints to communications [12]. The main goal of M2MC is to enable the sharing of information between electronic systems autonomously. Fig 2. shows the amalgation of machine to machine and IoT in real time. It finds its application in plethora of fields namely, transportation and logistics, medical and healthcare, insustrial and energy, security and survelliance. Machine to machine communication can happen using IoT in all these areas.

Nowadays, smartphones are equipped with a wide range of embedded sensors, with varied local and wide area connectivity capabilities, and thus they offer a unique opportunity to serve as mobile gateways for other more constrained devices with local connectivity. At the same time, they can gather context data about users and environment from the embedded sensors. These capabilities may be crucial for mobile M2M applications. The Internet of Things (IoT) with its unlimited range of applications that rely on everyday objects becoming intelligent connected devices is a major driver for M2M applications. The potential booming of M2M applications can exponentially increase the number and diversity of devices and traffic in the next years, which shall introduce further challenges to communications.

Current mobile M2M communications research focuses on performance evaluation and improvement, either in terms

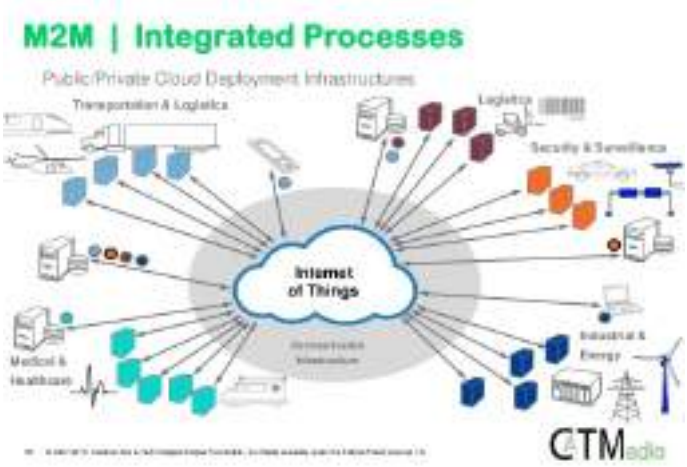

Fig. 2: Amalgamation of M2M and IoT

of delay or resource usage efficiency [15]. In other words Machine to Machine refers to technologies that allow both wireless and wired systems to communicate with other devices. M2M application protocols take a fundamental role in communication efficiency: protocol overheads, necessary number of management/control and information messages, reliability and security. All these impact the number and size of transmissions consequently, the energy and bandwidth consumptions in a mobile device. It finds it major application in protocols namely COAP, MQTT, AMQP, XMPP.

M2M technology mainly focusses on intellectually connecting the machine nodes or the smart objects. The smart objects are the sensors and the actuators in the IoT framework.The sensors need a mechanism to set up the design framework, to initialise the system configuaration for data transmission. The actuators require a technique or method to sens its control command to the designated recepient. The information overhead like the sensor identification, receiver identification, nuber of bytes of data should be carefully designed, given more attention to minimal data transfer. This will assist in better bandwidth utilisation for the internode communication in M2M technology. There should be a design platform to integrate instant messaging in the M2M communication. This designplatform is provided using the messaging protocols. The salient features of the various messaging methods are discussed next.

IV. THE MESSAGING PROTOCOLS

Instant messaging (IM) and Internet chat communication have seen enormous growth over the last several years. A number of key messaging technologies are emerging that will support the next generation of IoT applications, each of which can be used to connect devices in a distributed network. MQTT and CoAP address these needs through small message sizes, message management, and lightweight message overhead. Futhermore, the paper emphasis on other protocols namely XMPP, AMQP.

A. COAP

The Constrained Application Protocol (CoAP) is a synchronous request/response application layer protocol. CoAP aims to enable tiny devices with low power, computation and communication capabilities to utilize RESTful interactions. The development of the Constrained Application Protocol (CoAP) has made it possible to provide resource constrained devices with web service functionalities. CoAP is an HTTP like web transfer protocol able to extend the Representational State Transfer (REST) architecture to LoWPANs [2]. With the completion of the CoAP specification, it is expected that there will be millions of devices deployed in various application domains in the future. These applications range from smart energy, smart grid, building control, intelligent lighting control, industrial control systems, asset tracking, to environment monitoring.

CoAP would become the standard protocol to enable interaction between devices and to support IoT applications. CoAP is a binary protocol that runs over UDP. A messaging sub-layer adds a thin control layer that provides duplicate detection and optionally reliable delivery of messages based on a simple stop-and-wait mechanism for retransmissions [3]. CoAP is a request/response protocol that utilizes both synchronous and asynchronous responses. The reason for designing a UDP-based application layer protocol to manage the resources is to remove the TCP overhead and reduce bandwidth requirements [4]. Even though CoAP was created for the IoT and for M2M communications, it does not include any built-in security features.

B. MQTT

Message Queue Telemetry Transport (MQTT) was released by IBM and targets lightweight M2M communications. It is an asynchronous publish/subscribe protocol that runs on top of the TCP stack. MQTT aims at connecting embedded devices and networks with applications and middleware. MQTT utilizes the publish/subscribe pattern to provide transition flexibility and simplicity of implementation. The MQTT protocol represents an ideal messaging protocol for the IoT and M2M communications and is able to provide routing for small, cheap, low power and low memory devices in vulnerable and low bandwidth networks. In MQTT there is a broker (server) [8] that contains topics. Each client can be a publisher that sends information to the broker at a specific topic or/and a subscriber that receives automatic messages every time there is a new update in a topic one is subscribed [5].

MQTT is very lightweight and thus suited for M2M (Mobile to Mobile), WSN (Wireless Sensor Networks) and ultimately IoT (Internet of Things) scenarios where sensor and actor nodes communicate with applications through the

MQTT message broker. Recent years have witnessed the emergence of machine-to-machine (M2M) networks as an efficient means for providing automated communications among distributed devices [6]. The protocol was designed specifically for remote telemetry applications, with three specific design goals: (1) it should offer a once-and-once-only assured delivery mode to enable a message to be reliably transferred all the way from a remote sensor to a back-end application.(2) The protocol should be as lightweight as possible across the ”wire” (or other communication medium) most remote telemetry is done over low bandwidth, high cost networks, and so minimizing the overhead of each message is highly desirable. (3) the protocol should be very easy to implement on embedded devices such as sensors and gateways.

In conclusion, MQTT is a message-centric wire protocol designed for M2M communications that enables the transfer of telemetry-style data in the form of messages from devices, along high latency or constrained networks, to a server or small message broker. Devices may range from sensors and actuators, to mobile phones, embedded systems on vehicles, or laptops and full scale computers. It supports publish-and-subscribe style communications and is extremely simple.

C. XMPP

The Extensible Messaging and Presence Protocol (XMPP) was designed for chatting and message exchanging. It was standardized by the IETF over a decade ago, thus being a well-proven protocol that has been used widely all over the Internet. Google stopped supporting the XMPP standard due to the lack of worldwide support. XMPP supports the publish/subscribe architecture that is more suitable for the IoT in contrast to CoAPs request/response approach. Furthermore, it is an already established protocol that is supported all over the Internet as a plus with regard to the relatively new MQTT [7]. XMPP is an IETF instant messaging (IM) standard that is used for multi-party chatting, voice and video calling and telepresence. [8]. Many XMPP features make it a preferred protocol by most IM applications and relevant within the scope of the IoT. It runs over a variety of Internet-based platforms in a decentralized fashion. XMPP is secure and allows for the addition of new applications on top of the core protocols.

D. AMQP

The Advanced Message Queuing Protocol (AMQP) is a protocol that arose from the financial industry. It can utilize different transport protocols but it assumes an underlying reliable transport protocol such as TCP. AMQP provides asynchronous publish/subscribe communication with messaging. Its main advantage is its store-and-forward feature that ensures reliability even after network disruptions [9]. It supports reliable communication via message delivery guarantee primitives including at-most-once, at-least-once and exactly once delivery. AMQP requires a reliable transport protocol like TCP to exchange messages. AMQP defines a layer of messaging on top of its transport layer. Messaging capabilities are handled in this layer. By defining a wire-level protocol, AMQP implementations are able to interoperate with each other.

TABLE II: Comparison of Messaging Protocols

	MQTT	COAP	XMPP
Mode	Publish/Subscribe	Client/Server	Client/Server
Transport	TCP or UDP	UDP	TCP or HTTP
Security	SSL/TLS	DTLS	SSL/TLS

The above table summarises the mode, transport and security of the three messaging protocols, MQTT, COAP and XMPP. Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), both frequently referred to as “SSL”, are cryptographic protocols that provide communications security over a computer network. Both MQTT and XMPP adapt this secure connection system to ensure reliable security. At its core, TLS and SSL are cryptographic protocols which use a handshake mechanism to negotiate various parameters to create a secure connection between the client and the server. MQTT relies on TCP as transport protocol, which means by default the connection does not use an encrypted communication. To encrypt the whole MQTT communication, most of the MQTT brokers allow the use of TLS instead of plain TCP. The same is the case with XMPP protocol.

On the other hand, In information technology, the Datagram Transport Layer Security (DTLS) communications protocol provides communications security for datagram protocols. DTLS allows datagram-based applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery. COAP being a data gram protocol uses DTLS security. To protect the transmission of confidential information secure CoAP uses datagram transport layer security (DTLS) as the security protocol for communication and authentication of communicating devices. DTLS was initially designed for powerful devices that are connected through reliable and high bandwidth link.

V. LoRA - LoRAWAN PROTOCOL

LoRa is a long-range, low-power, low-bitrate, wireless telecommunications system, promoted as an infrastructure solution for the Internet of Things. This system aims at being usable in long-lived battery-powered devices, where the energy consumption is of paramount importance. The long-range and low-power nature of LoRa makes it an interesting candidate for smart sensing technology in civil infrastructures (such as health monitoring, smart metering, environment monitoring, etc.), as well as in industrial application. Smart technologies have improved the way we interact and are

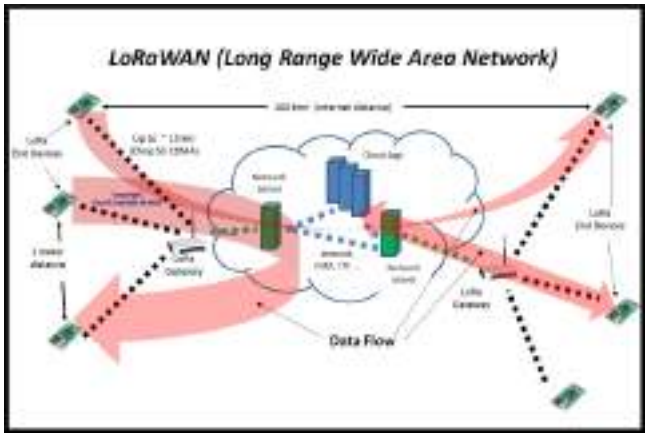

Fig. 3: LoRA and its establishment

addressing some of the biggest challenges faced by cities and communities: climate change, pollution control, early warning of natural disasters and saving lives. This wireless technology is being integrated into cars, street lights, manufacturing equipment, home appliances, and wearable devices [13].

LoRa is a physical layer specification based on CSS with integrated Forward Error Correction (FEC). Transmissions use a wide band to counter interference and to handle frequency offsets due to low cost crystals. A LoRa receiver can decode transmissions 19.5 dB below the noise floor. Thus, very long communication distances can be bridged. LoRa key properties are: long range, high robustness, multipath resistance, Doppler resistance, low power. Using a LoRa radio in a sensor network has some interesting aspects. First, since the range is relatively large (hundreds of meter indoors, kilometres outdoors), networks can span large areas without routing over many hops. In many cases one hop from every node to the sink is feasible. Secondly, transmission on the same carrier frequency, but with different spreading factor, are orthogonal. This creates the opportunity of dividing the channel in virtual subchannels. Thirdly, when transmissions occur at the same time with the same parameters, the strongest transmission will be received with high probability. Concurrent transmissions are nondestructive even when their contents is different. Fig 3. demonstrates the LoRAWAN features with its range of communication networking.

LoRA has a special benefit of inter device communication and gateway communication. There is a configuaration available to perform device to device communication in the vicinity of 1m LoRA end devices. The LoRA gateway feature provides a means of communication to the transmitter network server, cloud server and the receiver network server. From the receiver network server data can be transferred via the LoRA gateway to the receiver end devices. The long range distance that can be established in such LoRAWAN setup is 100 to 150 kilometers.

VI. CONCLUSION

The emerging idea of the Internet of Things (IoT) is rapidly finding its path throughout our modern life, aiming to improve the quality of life by connecting many smart devices, technologies, and applications. Overall, the IoT would allow for the automation of everything around us. The amalgamation of M2M communications and IoT is addressed. In conclusion, this paper presented an overview of the premise of this concept, its network communications, protocols and the recent research addressing different aspects of the IoT. The data transfer protocols for instant messaging are compared based on their mode, transport and security.

REFERENCES

[1] Tasos Kaukalias and Periklis Chatzimisios, “Internet of Things (IoT) C Enabling technologies, applications and open issues,” Encyclopedia of Information Science and Technology (3rd Ed.), IGI Global Press, August 2014, pp. 134 - 136.

[2] W. Colitti, K. Steenhaut, N. De Caro, B. Buta and V. Dobrota, “Evaluation of constrained application protocol for wireless sensor networks in Local Metropolitan Area Networks (LANMAN),” 18th IEEE Workshop , Jan 2011, pp. 1-6.

[3] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, “ Hypertext Transfer Protocol HTTP,” IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), June 1999, pp. 45 - 46.

[4] Sye Loong Keoh, Sandeep S. Kumar, Hannes Tschofenig, “Securing the Internet of Things: A Standardization Perspective,” Internet of Things Journal IEEE (Volume: 1, Issue: 3), June 2014, pp. 265-275.

[5] Shinho Lee, Hyeonwoo Kim, Dong-kweon Hong, Hongtaek Ju, “Correlation Analysis of MQTT Loss and Delay According to QoS Level,” International Conference on Information Networking (ICOIN), 28-30 Jan. 2013, pp. 714-717.

[6] Min Chen, Jiafu Wan, Gonzalez, S., Xiaofei Liao, Leung, V.C.M., “A Survey of Recent Developments in Home M2M Networks,” Communications Surveys and Tutorials, IEEE , vol.16, no.1, First Quarter 2014, pp. 98,114.

[7] Michael Kirsche, Ronny Klauck, Unify to Bridge Gaps, “Bringing XMPP into the Internet of Things,” IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), 19-23 March 2012, pp. 455-458.

[8] P. Saint-Andre, “Extensible messaging and presence protocol (XMPP): Core, Internet Engineering Task Force (IETF), and Request for Comments,” IEEE International Conference on Distributed Computing in Sensor Systems, March 2011, pp. 230 - 234.

[9] Frank T. Johnsen, Trude H. Bloebaum, Morten Avlesen, Skage Spjelkavik, Bjrn Vik, “Evaluation of Transport Protocols for Web Services,” Military Communications and Information Systems Conference (MCC), 7-9 Oct. 2013, pp. 54 - 56.

[10] Michael Hausenblas, “Smart phones and Internet of Things,” Chief Engineer, EMEA, MapR, Popular Blog on loT, Nov. 2014, pp 1-4.

[11] G. M. Leeet al.,“The loT-Concept and Problem Statement,” IETF Standard draft-lee-iot-problem-statement-05, Jul. 30, 2012, pp. 67 - 68.

[12] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” Com put. Newt. Vol. 54, No. 15, Oct. 2010, pp. 2787-2805.

[13] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, ”Internet of Things: Vision, applications and research challenges,” Ad Hoc Netw., Vol. 10, No. 7, Sep. 2012, pp. 1497-1516.

[14] Giusto D., Iera A., Morabito G., Atzori L. “The Internet of Things,” Springer-Verlag Berlin/Heidelberg, Germany, Jan 2010, pp. 67 - 71.

[15] Zhang Y., Yu R., Xie S., Yao W., Xiao Y., Guizani M. “Home M2M networks: Architectures, standards, and QoS improvement, IEEE Conference May 2011, pp.44 52.

[16] Hussein D., Han S. N., Han X., Lee G. M., Crespi N. “A framework for social device networking Proceedings,” 9th IEEE International Conference on Distributed Computing in Sensor Systems, May 2013 Cambridge, pp. 45 - 48.